Techniques of Water-Resources Investigations of the United States Geological Survey

Chapter B3

TYPE CURVES FOR SELECTED PROBLEMS OF FLOW TO WELLS IN CONFINED AQUIFERS

By J. E. Reed

Book 3
APPLICATIONS OF HYDRAULICS

UNITED STATES DEPARTMENT OF THE INTERIOR

CECIL D. ANDRUS, Secretary

GEOLOGICAL SURVEY

H. William Menard, Director

UNITED STATES GOVERNMENT PRINTING OFFICE : 1980

For sale by the Branch of Distribution, U.S. Geological Survey,
1200 South Eads Street, Arlington, VA 22202

PREFACE

The series of manuals on techniques describes procedures for planning and executing specialized work in water-resources investigations. The material is grouped under major subject headings called books and further subdivided into sections and chapters; section B of book 3 is on ground-water techniques.

Provisional drafts of chapters are distributed to field offices of the U.S. Geological Survey for their use. These drafts are subject to revision because of experience in use or because of advancement in knowledge, techniques, or equipment. After the technique described in a chapter is sufficiently developed, the chapter is published and is sold by the U.S. Geological Survey, 1200 South Eads Street, Arlington, VA 22202 (authorized agent of Superintendent of Documents, Government Printing Office).

TECHNIQUES OF WATER-RESOURCES INVESTIGATIONS OF THE U.S. GEOLOGICAL SURVEY

The U.S. Geological Survey publishes a series of manuals describing procedures for planning and conducting specialized work in water-resources investigations. The manuals published to date are listed below and may be ordered by mail from the **Branch of Distribution, U.S. Geological Survey, 1200 South Eads Street, Arlington, VA 22202** (an authorized agent of the Superintendent of Documents, Government Printing Office).

Prepayment is required. Remittances should be sent by check or money order payable to U.S. Geological Survey. Prices are not included in the listing below as they are subject to change. **Current prices can be obtained by calling the USGS Branch of Distribution, phone (202) 751-6777.** Prices include cost of domestic surface transportation. For transmittal outside the U.S.A. (except to Canada and Mexico) a surcharge of 25 percent of the net bill should be included to cover surface transportation.

When ordering any of these publications, please give the title, book number, chapter number, and "U.S. Geological Survey Techniques of Water-Resources Investigations."

TWI 1-D1. Water temperature–influential factors, field measurement, and data presentation, by H. H. Stevens, Jr., J. F. Ficke, and G. F. Smoot. 1975. 65 pages.

TWI 1-D2. Guidelines for collection and field analysis of ground-water samples for selected unstable constituents, by W. W. Wood. 1976. 24 pages.

TWI 2-D1. Application of surface geophysics to ground-water investigations, by A. A. R. Zohdy, G. P. Eaton, and D. R. Mabey. 1974. 116 pages.

TWI 2-E1. Application of borehole geophysics to water-resources investigations, by W. S. Keys and L. M. MacCary. 1971. 126 pages.

TWI 3-A1. General field and office procedures for indirect discharge measurements, by M. A. Benson and Tate Dalrymple. 1967. 30 pages.

TWI 3-A2. Measurement of peak discharge by the slope-area method, by Tate Dalrymple and M. A. Benson. 1967. 12 pages.

TWI 3-A3. Measurement of peak discharge at culverts by indirect methods, by G. L. Bodhaine. 1968. 60 pages.

TWI 3-A4. Measurement of peak discharge at width contractions by indirect methods, by H. F. Matthai. 1967. 44 pages.

TWI 3-A5. Measurement of peak discharge at dams by indirect methods, by Harry Hulsing. 1967. 29 pages.

TWI 3-A6. General procedure for gaging streams, by R. W. Carter and Jacob Davidian. 1968. 13 pages.

TWI 3-A7. Stage measurements at gaging stations, by T. J. Buchanan and W. P. Somers. 1968. 28 pages.

TWI 3-A8. Discharge measurements at gaging stations, by T. J. Buchanan and W. P. Somers. 1969. 65 pages.

TWI 3-A11. Measurement of discharge by moving-boat method, by G. F. Smoot and C. E. Novak. 1969. 22 pages.

TWI 3-B1. Aquifer-test design, observation, and data analysis, by R. W. Stallman. 1971. 26 pages.

TWI 3-B2. Introduction to ground-water hydraulics, a programed text for self-instruction, by G. D. Bennett. 1976. 172 pages.

TWI 3-B3. Type curves for selected problems of flow to wells in confined aquifers, by J. E. Reed.

TWI 3-C1. Fluvial sediment concepts, by H. P. Guy. 1970. 55 pages.

TWI 3-C2. Field methods for measurement of fluvial sediment, by H. P. Guy and V. W. Norman. 1970. 59 pages.

READEX FICHE best copy available

TWI 3-C3. Computation of fluvial-sediment discharge, by George Porterfield. 1972. 66 pages.
TWI 4-A1. Some statistical tools in hydrology, by H. C. Riggs. 1968. 39 pages.
TWI 4-A2. Frequency curves, by H. C. Riggs. 1968. 15 pages.
TWI 4-B1. Low-flow investigations, by H. C. Riggs. 1972. 18 pages.
TWI 4-B2. Storage analyses for water supply, by H. C. Riggs and C. H. Hardison. 1973. 20 pages.
TWI 4-B3. Regional analyses of streamflow characteristics, by H. C. Riggs. 1973. 15 pages.
TWI 4-D1. Computation of rate and volume of stream depletion by wells, by C. T. Jenkins. 1970. 17 pages.
TWI 5-A1. Methods for determination of inorganic substances in water and fluvial sediments, by M. W. Skougstad and others, editors. 1979. 626 pages.
TWI 5-A2. Determination of minor elements in water by emission spectroscopy, by P. R. Barnett and E. C. Mallory, Jr. 1971. 31 pages.
TWI 5-A3. Methods for analysis of organic substances in water, by D. F. Goerlitz and Eugene Brown. 1972. 40 pages.
TWI 5-A4. Methods for collection and analysis of aquatic biological and microbiological samples, edited by P. E. Greeson, T. A. Ehlke, G. A. Irwin, B. W. Lium, and K. V. Slack. 1977. 332 pages.
TWI 5-A5. Methods for determination of radioactive substances in water and fluvial sediments, by L. L. Thatcher, V. J. Janzer, and K. W. Edwards. 1977. 95 pages.
TWI 5-C1. Laboratory theory and methods for sediment analysis, by H. P. Guy. 1969. 58 pages.
TWI 7-C1. Finite difference model for aquifer simulation in two dimensions with results of numerical experiments, by P. C. Trescott, G. F. Pinder, and S. P. Larson. 1976. 116 pages.
TWI 7-C2. Computer model of two-dimensional solute transport and dispersion in ground water, by L. F. Konikow and J. D. Bredehoeft. 1978. 90 pages.
TWI 7-C3. A model for simulation of flow in singular and interconnected channels, by R. W. Schaffranek, R. A. Baltzer, and D. E. Goldberg.
TWI 8-A1. Methods of measuring water levels in deep wells, by M. S. Garber and F. C. Koopman. 1968. 23 pages.
TWI 8-B2. Calibration and maintenance of vertical-axis type current meters, by G. F. Smoot and C. E. Novak. 1968. 15 pages.

CONTENTS

	Page
Abstract	1
Introduction	1
Summaries of type-curve solutions for confined ground-water flow toward a well in an infinite aquifer	5
Solution 1: Constant discharge from a fully penetrating well in a nonleaky aquifer (Theis equation)	5
Solution 2: Constant discharge from a partially penetrating well in a nonleaky aquifer	8
Solution 3: Constant drawdown in a well in a nonleaky aquifer	13
Solution 4: Constant discharge from a fully penetrating well in a leaky aquifer	18
Solution 5: Constant discharge from a well in a leaky aquifer with storage of water in the confining beds	25
Summaries of type-curve solutions, etc.—Continued	Page
Solution 6: Constant discharge from a partially penetrating well in a leaky aquifer	29
Solution 7: Constant drawdown in a well in a leaky aquifer	34
Solution 8: Constant discharge from a fully penetrating well of finite diameter in a nonleaky aquifer	37
Solution 9: Slug test for a finite diameter well in a nonleaky aquifer	45
Solution 10: Constant discharge from a fully penetrating well in an aquifer that is anisotropic in the horizontal plane	46
Solution 11: Variable discharge from a fully penetrating well in a leaky aquifer	49
References	52

ILLUSTRATIONS

		Page
PLATE	1. Type-curve solutions for confined ground-water flow toward a well in an infinite aquifer	In pocket
FIGURE	0.1. Graph showing the relation of $1/u$, $W(u)$ type curve and t,s data plot	2
	0.2. Graph showing the application of the principle of superposition to aquifer tests	3
	1.1. Cross section through a discharging well in a nonleaky aquifer	6
	1.2. Graph showing type curve of dimensionless drawdown ($W(u)$) versus dimensionless time ($1/u$) for constant discharge from an artesian well (Theis curve)	Plate 1
	2.1. Cross section through a discharging well that is screened in a part of a nonleaky aquifer	9
	2.2. Graph showing four selected type curves of dimensionless drawdown ($W(u)+f$) versus dimensionless time ($1/u$) for constant discharge from a partially penetrating artesian well	Plate 1
	2.3. Graph of the drawdown correction factor f_s versus ar/b	10
	2.4. Example of output from program for partial penetration in a nonleaky artesian aquifer	17
	3.1. Cross section through a well with constant drawdown in a nonleaky aquifer	18
	3.2. Graph showing type curve of dimensionless discharge ($G(\alpha)$) versus dimensionless time (α) for constant drawdown in an artesian well	Plate 1
	3.3. Graph showing type curves of dimensionless drawdown ($A(\tau,\rho)$) versus dimensionless time (τ/ρ^2) for constant drawdown in a well in a nonleaky aquifer	Plate 1
	4.1. Cross section through a discharging well in a leaky aquifer	21
	4.2. Graph showing type curve of $L(u,v)$ versus $1/u$	Plate 1
	4.3. Graph showing type curve of the Bessel function $K_o(x)$ versus x	Plate 1
	4.4. Example of output from program for computing drawdown due to constant discharge from a well in a leaky artesian aquifer	24
	5.1. Cross sections through discharging wells in leaky aquifers with storage of water in the confining beds, illustrating three different cases of boundary conditions	27
	5.2. Graph showing dimensionless drawdown ($H(u, \beta)$) versus dimensionless time ($1/u$) for a well fully penetrating a leaky artesian aquifer with storage of water in leaky confining beds	Plate 1

Page

FIGURE 5.3. Example of output from program for computing drawdown due to constant discharge from a well in a leaky aquifer with storage of water in the confining beds ---- 30
6.1. Cross section through a discharging well that is screened in part of a leaky aquifer ---- 32
6.2. Eight selected type curves of dimensionless drawdown ($W(u,\beta)+f$) versus dimensionless time ($1/u$) for constant discharge from a partially penetrating well in a leaky artesian aquifer Plate 1
6.3. Example of output from program for partial penetration in a leaky artesian aquifer ---- 33
7.1. Cross section through a well with constant drawdown in a leaky aquifer ---- 35
7.2. Graph showing type curve of dimensionless discharge ($G(\alpha,r_w/B)$) versus dimensionless time (α) for constant drawdown in a well in a leaky aquifer ---- Plate 1
7.3. Graph showing ten selected type curves of dimensionless drawdown versus dimensionless time for constant drawdown in a well in a leaky aquifer ---- Plate 1
7.4. Example of output from program for constant drawdown in a well in a leaky artesian aquifer 38
8.1. Cross section through a discharging well of finite diameter ---- 39
8.2. Graph showing five selected type curves of $F(u_w,\alpha)$, and the Theis solution, versus $1/u_w$ ---- 40
8.3. Graph showing eight selected type curves of $F(u,\alpha,\rho)$ for $\alpha=10^{-4}$, and the Theis solution, versus $1/u$ ---- Plate 1
8.4. Example of output from program for drawdown inside a well of finite diameter due to constant discharge ---- 44
8.5. Example of output from program for drawdown outside a well of finite diameter due to constant discharge ---- 44
9.1. Cross section through a well in which a slug of water is suddenly injected ---- 46
9.2. Graph showing ten selected type curves of $F(\beta,\alpha)$ versus β ---- Plate 1
9.3. Example of output from program to compute change in water level due to sudden injection of a slug of water into a well ---- 48
10.1. Plan view showing coordinate axes ---- 49
11.1. Cross section through a well with variable discharge ---- 51
11.2. Graph showing eleven selected type curves for linearly varying discharge, $Q(t)=Q_0(1+ct)$ ---- Plate 1
11.3. Example of output from program to compute the convolution integral for a leaky aquifer ---- 53

TABLES

Page

TABLE 1.1. Values of Theis equation $W(u)$ for values of $1/u$ ---- 7
2.1. Listing of program for partial penetration in a nonleaky artesian aquifer ---- 57
3.1. Values of $G(\alpha)$ ---- 19
3.2. Values of $A(\tau,\rho)$ ---- 20
4.1. Selected values of $W(u,r/B)$ ---- 22
4.2. Selected values of $K_0(x)$ ---- 23
4.3. Listing of program for radial flow in a leaky artesian aquifer ---- 66
5.1. Values of $H(u,\beta)$ for selected values of u and β ---- 28
5.2. Listing of program for radial flow in a leaky artesian aquifer with storage of water in the confining beds ---- 72
6.1. Listing of program for partial penetration in a leaky artesian aquifer ---- 75
7.1. Values of $G(\alpha,r_w/B)$ ---- 36
7.2. Listing of program for constant drawdown in a well in an infinite leaky aquifer ---- 83
8.1. Values of the function $F(u,\alpha,\rho)$ ---- 41
8.2. Listing of programs for constant discharge from a fully penetrating well of finite diameter ---- 91
9.1. Values of H/H_0 ---- 47
9.2. Listing of program to compute change in water level due to sudden injection of a slug of water into a well ---- 99
11.1. Listing of program to compute the convolution integral for a leaky aquifer ---- 102

SYMBOLS AND DIMENSIONS

[Numbers in parentheses indicate the solutions to which the definition applies. If no number appears, the symbol has only one definition in this report]

Symbol	Dimension	Description
a	Dimensionless	$\sqrt{K_z/K_r}$.
b	L	Aquifer thickness.
b'	L	Thickness of confining bed (4, 6, 7, 11); specifically the upper confining bed (5).
b''	L	Thickness of lower confining bed.
d	L	Depth from top of aquifer to top of pumped well screen.
d'	L	Depth from top of aquifer to top of observation-well screen.
H	L	Change in water level in well.
H_0	L	Initial head increase in well.
h	L	Change in water level in aquifer.
K	LT^{-1}	Hydraulic conductivity of aquifer.
K_r	LT^{-1}	Hydraulic conductivity of the aquifer in the radial direction.
K_z	LT^{-1}	Hydraulic conductivity of the aquifer in the vertical direction.
K'	LT^{-1}	Hydraulic conductivity of confining bed (4, 6, 7); specifically the upper confining bed (5).
K''	LT^{-1}	Hydraulic conductivity of lower confining bed.
l	L	Depth from top of aquifer to bottom of pumped well screen.
l'	L	Depth from top of aquifer to bottom of observation-well screen.
Q	L^3T^{-1}	Discharge rate.
$Q(t)$	L^3T^{-1}	Discharge rate.
r	L	Radial distance from center of pumping, flowing, or injecting well.
r_c	L	Radius of well casing or open hole in the interval where the water level changes.
r_w	L	Effective radius of well screen or open hole for pumping, flowing, or injecting well.
S	Dimensionless	Storage coefficient.
S_s	L^{-1}	Specific storage of aquifer.
S_s'	L^{-1}	Specific storage of confining beds.
S'	Dimensionless	Storage coefficient of upper confining bed.
S''	Dimensionless	Storage coefficient of lower confining bed.
s	L	Drawdown in head (change in water level).
s_1	L	Drawdown in upper confining bed.
s_2	L	Drawdown in lower confining bed.
s_w	L	Constant drawdown in discharging well.
T	L^2T^{-1}	Transmissivity.
T_{xx}, T_{xy}, T_{yy}	L^2T^{-1}	Components of the transmissivity tensor in any orthogonal x-, y-axis system.
$T_{\epsilon\epsilon}$, $T_{\eta\eta}$	L^2T^{-1}	Transmissivities along two principal axes, ϵ and η, such that $T_{\epsilon\eta}=0$.
t	T	Time.
t'	Dimensionless	Variable of integration.
u	Dimensionless	$r^2S/4Tt$(2, 6); variable of integration (3, 7, 9).
v	Dimensionless	Variable of integration.
x	Dimensionless	Dummy variable (2, 5); variable of integration (3).
x, y	L	Distances from the pumped well for an arbitrary rectangular coordinate system (10).
y	Dimensionless	Variable of integration (1, 2, 4, 5, 6).
z	L	Depth from top of aquifer, also, specifically, the depth to bottom of a piezometer (2, 6); depth below top of upper confining bed (5).
z	Dimensionless	Dummy variable (10).
α	Dimensionless	Tt/Sr_w^2.
β	Dimensionless	Variable of integration.
θ	Dimensionless	Angle between x axis and ϵ axis.
ϵ, η	L	Distances from pumped well in a coordinate system colinear with principal axes of transmissivity tensor.
ρ	Dimensionless	r/r_w.
τ	Dimensionless	Tt/Sr_w^2.

TYPE CURVES FOR SELECTED PROBLEMS OF FLOW TO WELLS IN CONFINED AQUIFERS

By J. E. Reed

Abstract

This report presents type curves and related material for 11 conditions of flow to wells in confined aquifers. These solutions, compiled from hydrologic literature, span an interval of time from Theis (1935) to Papadopulos, Bredehoeft, and Cooper (1973). Solutions are presented for constant discharge, constant drawdown, and variable discharge for pumping wells that fully penetrate leaky and nonleaky aquifers. Solutions for wells that partially penetrate leaky and nonleaky aquifers are included. Also, solutions are included for the effect of finite well radius and the sudden injection of a volume of water for nonleaky aquifers. Each problem includes the partial differential equation, boundary and initial conditions, and solutions. Programs in FORTRAN for calculating additional function values are included for most of the solutions.

Introduction

The purpose of this report is to assemble, under one cover and in a standard format, the more commonly used type-curve solutions for confined ground-water flow toward a well in an infinite aquifer. Some of these solutions are only published in several different journals; some of these journals are not readily obtainable. Other solutions which are included in several references (for example, Ferris and others, 1962; Walton, 1962; Hantush, 1964a; Lohman, 1972) are included here for completeness.

The need for a compendium of type curves for aquifer-test analysis was recognized by Robert W. Stallman, who initiated the work on it. However, ill health and the press of other duties prevented him from personally carrying out his concept, but he never ceased to advocate the need for the compendium. Although it is reduced in scope from his original concept, this report should be recognized to be a result of Stallman's foresight and endeavors in the field of ground-water hydrology.

The type-curve method was devised by C. V. Theis (Wenzel, 1942, p. 88) to determine the two unknown parameters, S and T, in the equations

$$s = (Q/4\pi T)W(u)$$

and

$$u = r^2S/(4Tt),$$

where s is the drawdown in water level in response to the pumping rate Q in an aquifer with transmissivity T and storage coefficient S. The distance r from the pumping well, and the elapsed time t since pumping began, combine with S and T to define a dimensionless variable u and corresponding dimensionless response $W(u)$. Briefly, the method consists of plotting a function curve or type curve, such as $(1/u, W(u))$ on logarithmic-scale graph paper, and plotting the time-drawdown $(t\text{-}s)$ data on a second sheet having the same scales. This is equivalent to expressing the preceding equations as

$$\log s = \log Q/4\pi T + \log W(u)$$

and

$$\log 1/u = \log t + \log 4T/r^2S.$$

If the two sheets are superimposed and matched, keeping coordinate axes parallel, as shown in figure 0.1, the respective coordinate

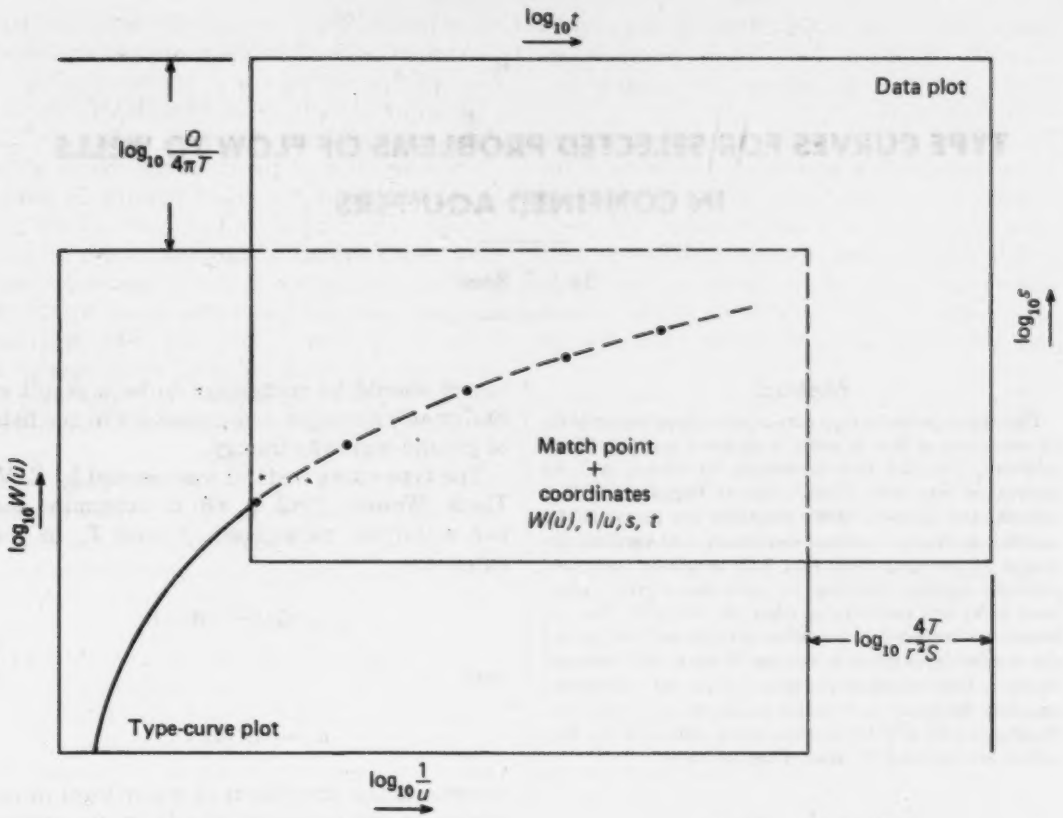

FIGURE 0.1.—Relation of $1/u, W(u)$ type curve and t, s data plot. Modified from Stallman (1971, p. 5, fig. 1).

axes will be related by constant factors: $s/W(u)=C_1$ and $t/(1/u)=C_2$. The values of these two constants are

$$C_1=Q/(4\pi T)$$

and

$$C_2=r^2S/(4T).$$

Thus, a common match point for the two curves may be chosen, and the four coordinate points—$W(u)$, $1/u$, s, and t—recorded for the common match point. T can be obtained from the equation $T=QW(u)/(4\pi s)$, and then S can be solved from the equation $S=4Tut/r^2$, where $W(u)$, $1/u$, s, and t are the match-point values.

It is apparent that the type curves, and data, can be plotted in several ways. That is, the function curve, using $W(u)$ as an example, could be plotted as $(u,W(u))$ with corresponding data plots of $(1/t,s)$ or $(r^2/t,s)$; or could be plotted as $(1/u,W(u))$ with corresponding data plots of (t,s) or $(t/r^2,s)$. The type-curve method is covered more fully by Ferris, Knowles, Brown, and Stallman (1962, p. 94).

The type curves presented in this report are shown on two different plots. One plot has both logarithmic scales with 1.85 inches per log-cycle, such as K and E 467522.[1] The other plot is arithmetic-logarithmic scale with the logarithmic scale 2 inches per log-cycle and the arithmetic scale with divisions at multiples of 0.1, 0.5, and 1.0 inches, such as K and E 466213.

Other methods exist for analysis of aquifer-test data. Among them are methods based on plots of data on semi-log paper, developed by

[1]The use of brand names in this report is for identification purposes only and does not imply endorsement by the U.S. Geological Survey.

Jacob (Ferris and others, 1962, p. 98) and by Hantush (1956, p. 703). These methods are useful, but they are beyond the scope of this report.

Aquifer tests deal with only one component of the natural flow system. The isolation of the effects of one stress upon the system is based upon the technique of superposition. This technique requires that the natural flow system can be approximated as a linear system, one in which total flow is the addition of the individual flow components resulting from distinct stresses.

The use of the principle of superposition is implied in most aquifer-test analyses. The term "superposition," as here applied, is derived from the theory of linear differential equations. If the partial-differential equation is linear (in the dependent variable and its derivatives), two or more solutions, each for a given set of boundary and initial conditions, can be summed algebraically to obtain a solution for the combined conditions. For instance, consider a situation (fig. 0.2) where a well has been pumping for some time at a constant rate Q_0, and the drawdown trend for that pumping rate has been established. Assume that the pumping rate increases by some amount ΔQ at some time t_1. Then the drawdown for that step increase in rate will be the change in drawdown from that occurring due to the pumpage Q_0.

Programs, written in FORTRAN, for calculating additional function values are included for most of the solutions. Some of the type-curve solutions would require an unreasonably long tabulation to include all the possible combinations of parameters. An alternative to a tabulation is the computer program that can calculate type-curve values for the parameters desired by the user. The programs could be easily modified to calculate aquifer response to more than one well, such as well fields or image-well systems (Ferris and others, 1962, p. 144). The programs have been tested and are probably reasonably free from error. However, because of the large number of possible parameter combinations, it was possible to test only a sample of possible parameter values. Therefore, errors might occur in future use of these programs.

"An aquifer test is a controlled field experiment made to determine the hydraulic properties of water-bearing and associated rocks" (Stallman, 1971). The areal variability of hydraulic properties in an aquifer limits aquifer tests to integrating these properties within the

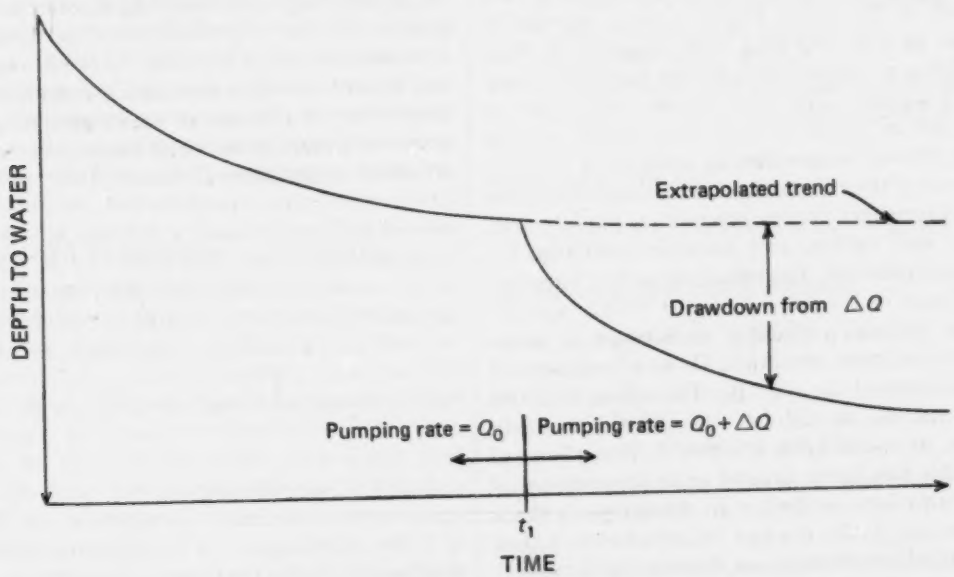

FIGURE 0.2.—The application of the principle of superposition to aquifer tests.

cone of depression produced during the test. Aquifer-test solutions are based on idealized representations of the aquifer, its boundaries, and the nature of the stress on the aquifer. The type-curve solutions presented in this report all have certain assumptions in common. The common assumptions are that the aquifer is horizontal and infinite in areal extent, that water is confined by less permeable beds above and below the aquifer, that the formation parameters are uniform in space and constant in time, that flow is laminar, and that water is released from storage instantaneously with a decline in head. Also implicit is the assumption that hydraulic potential or head is the only cause of flow in the system and that thermal, chemical, density, or other forces are not affecting flow. In addition to these common assumptions are special assumptions that characterize each solution summary. An important first step in aquifer-test analysis is deciding which simplified representations most closely match the usually complex field conditions.

Generally the best start in the analysis of aquifer-test data is with the most general set of type curves that apply to the situation, keeping in mind limitations of the method and effects that cause departures from the theoretical results. For example, the most general set of type curves for constant discharge presented in this report is for leaky aquifers with storage of water in the confining beds, *solution* 5. This includes, as a limiting case, the curve for a nonleaky aquifer. The most severe limitation on this set of curves is that they apply only at early times, as specified in *solution* 5.

Some of the effects that cause departure from the theoretical curves are partial penetration, finite well radius, and variable discharge for the pumped well. The effects of partial penetration must be considered when $r/b<1.5$, and because vertical-horizontal anisotropy is probably a common condition, these effects should be considered for $r/b<10$. The effect of finite well radius should be considered for early times, as specified in *solution* 8. The effects of variable discharge depend upon the manner of the variation. A change in discharge is more important if the change is monotonic, either continually increasing or decreasing. This fact is shown by the type curves for *solution* 11, where a monotonic change of 10 percent caused a significant departure from the Theis curve. If the discharge variation consists of random "noise" about a constant discharge, a 10-percent variation is not significant. The most general set of type curves for tests on flowing wells is *solution* 7, for leaky aquifers, which includes nonleaky aquifers as a limiting case. The only set of curves for slug tests is given in *solution* 9.

A recurring problem in type-curve solution for unknown hydrologic parameters is that of nonuniqueness. That is, function curves for different parameter values sometimes have similar shapes. An example of this is given by Stallman (1971, p. 19 and fig. 6). He indicated that the selection of the conceptual model is very important in interpreting the test results. Equally important is adequate testing of the conceptual model. Corroboration of the conceptual model is indicated by similar results for hydrologic parameters from data collected at varying distances from the pumped well, depths within the aquifer, and at different observation times. However, proof of suitability of the conceptual model ultimately rests on field investigations and not on curve matching.

As an example of similar curve shapes for different situations, consider the case of constant discharge in a nonleaky aquifer with exponentially varying thickness. The thickness, b, is equal to $b_0\exp[-2(X-X_0)/a]$, where b_0 and X_0 are the thickness and X-coordinate, respectively, at the site of the discharging well and a is a parameter. The drawdown for this situation is given by Hantush (1962, p. 1529):

$$s=(Q/4\pi Kb_0)\exp(r/a\cos\Theta)\,W(u,r/a),$$

where

$$W(u,\beta)=\int_u^\infty (\exp(-y-\beta^2/4y)/y)\,dy,$$

$$u=r^2S_s/4Kt,$$

Q is the discharge, r is the distance from the discharging well, Θ is the angle, with apex at the discharging well, between the observation

well and the positive X-axis, K is the hydraulic conductivity of the aquifer, and S_s is the specific storage coefficient of the aquifer. This solution is similar to the equation describing drawdown in a leaky artesian aquifer (Hantush, 1956, p. 702), which is

$$s = (Q/4\pi T)\, W(u,r/B),$$

with $T=Kb$, $B=\sqrt{Tb'/K'}$, and b' and K' are the thickness and hydraulic conductivity, respectively, of the leaky confining bed. The other symbols are used as above.

These two functions have the same shape when plotted on logarithmic paper, and drawdown resulting from one function could be matched to a type curve of the other function. Suppose, as an example, that the "observed data" are described by the function for the aquifer with exponentially changing thickness. Suppose, also, that the hydrologist is unaware of the variation in thickness and that the family of type curves for leaky aquifers without storage in the confining beds, *solution* 4, has been chosen for analysis of the "observed data." Matching the data plots to the type curves and solving for unknown parameters by the methods suggested in *solution* 4 gives for the ratio of K_a, the apparent hydraulic conductivity, to K, the true hydraulic conductivity, $K_a/K=\exp((r/a)\cos\Theta)$. The ratio would be close to one only in the vicinity of the discharging well. The diffusivity, K/S_s, would be determined correctly, but the apparent specific storage coefficient would have the same percentage error as the apparent hydraulic conductivity. Most important of all, the erroneous conclusion would be that the aquifer is leaky, with leakage parameter $B = \sqrt{Kbb'/K'} = a$. This somewhat contrived example illustrates a principle in the interpretation of aquifer-test data. Conclusions about the hydrologic constraints on the response of the aquifer to pumping should not be based on the shape of the data curves. Inferences may be made from these curves, but they must be verified by other hydrologic and geologic data. Therefore, proof of the suitability of the conceptual model must come from field investigations.

Many of the old reports of the U.S. Geological Survey contain references to the terms "coefficient of transmissibility" and "field coefficient of permeability." These terms, which were expressed in inconsistent units of gallons and feet, have been replaced by transmissivity and hydraulic conductivity (Lohman and others, 1972, p. 4 and p. 13). Transmissivity and hydraulic conductivity are not solely properties of the porous medium; they are also determined by the kinematic viscosity of the liquid, which is a function of temperature. Field determinations of transmissivity or hydraulic conductivity are made at prevailing field temperatures, and no corrections for temperature are made.

Summaries of Type-Curve Solutions for Confined Ground-Water Flow Toward a Well in an Infinite Aquifer

Solution 1: Constant discharge from a fully penetrating well in a nonleaky aquifer (Theis equation)

Assumptions:

1. Well discharges at a constant rate, Q.
2. Well is of infinitesimal diameter and fully penetrates the aquifer.
3. Aquifer is not leaky.
4. Discharge from the well is derived exclusively from storage in the aquifer.

Differential equation:

$$\partial^2 s/\partial r^2 + (1/r)\,(\partial s/\partial r) = (S/T)(\partial s/\partial t)$$

Boundary and initial conditions:

$$s(r,0) = 0,\ r \geq 0 \qquad (1)$$

$$s(\infty,t) = 0,\ t \geq 0 \qquad (2)$$

$$Q = \begin{cases} 0, t<0 \\ \text{constant} >0,\ t \geq 0 \end{cases} \qquad (3)$$

$$\lim_{r \to 0} r\frac{\partial s}{\partial r} = -\frac{Q}{2\pi T},\ t \geq 0 \qquad (4)$$

Equation 1 states that initially drawdown is zero everywhere in the aquifer. Equation 2

states that the drawdown approaches zero as the distance from the well approaches infinity. Equation 3 states that the discharge from the well is constant throughout the pumping period. Equation 4 states that near the pumping well the flow toward the well is equal to its discharge.

Solution (Theis, 1935):

$$s = \frac{Q}{4\pi T}\int_u^\infty \frac{e^{-y}}{y}\,dy$$

$$u = \frac{r^2 S}{4Tt},$$

where

$$\int_u^\infty \frac{e^{-y}}{y}\,dy = W(u) = -0.577216 - \log_e u + u - \frac{u^2}{2!\,2} + \frac{u^3}{3!\,3} - \frac{u^4}{4!\,4} + \cdots\cdots$$

Comments:

Assumptions made are applicable to artesian aquifers (fig. 1.1). However, the solution may be applied to unconfined aquifers if drawdown is small compared with the saturated thickness of the aquifer and if water in the sediments through which the water table has fallen is discharged instantaneously with the fall of the water table. According to assumption 2, this solution does not consider the effect of the change in storage within the pumping well. Assumption 2 is acceptable if

$$t > 2.5\times 10^2 r_c^2/T$$

(Papadopulos and Cooper, 1967, p. 242), where r_c is the radius of the well casing in the interval over which the water-level declines, and other symbols are as defined previously. Figure 1.2 on plate 1 is a logarithmic graph of $W(u)=4\pi sT/Q$ plotted on the vertical coordinates versus $1/u=4Tt/(r^2S)$ plotted on the horizontal coordinates. The test data should be plotted with s on the vertical coordinates and corresponding values of t or t/r^2 on the horizontal coordinates.

Values of $W(u)$ for u between 0 and 170 may be computed by using subroutine EXPI of the IBM System/360 Scientific Subroutine Package. Table 1.1 gives values of $W(u)$ for selected values of $1/u$ between 1×10^{-1} and 9×10^{14}, as calculated by this subroutine.

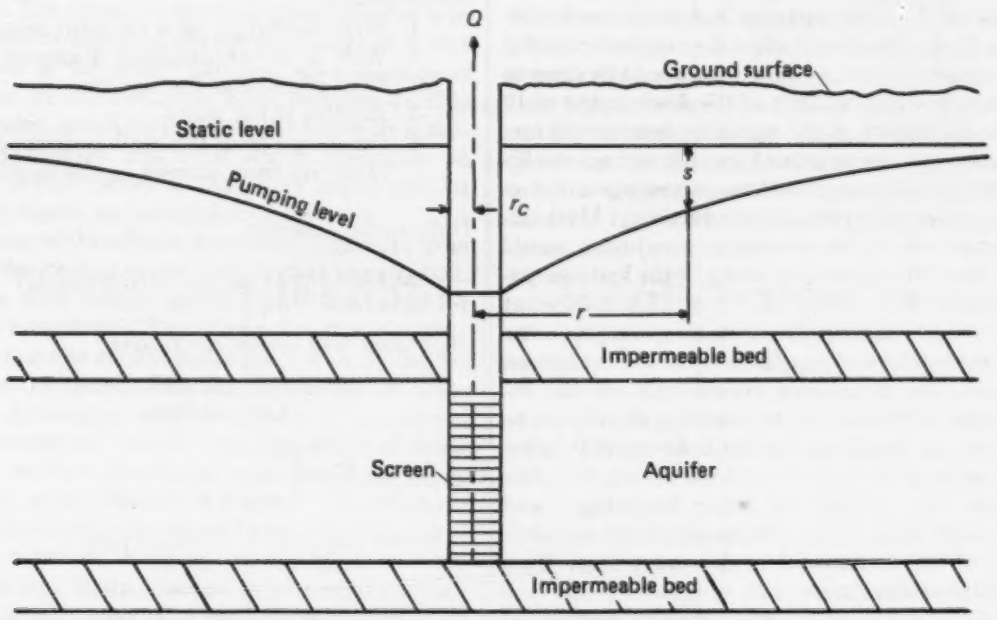

FIGURE 1.1.—Cross section through a discharging well in a nonleaky aquifer.

TABLE 1.1.—*Values of Theis equation W(u) for values of 1/u*

$1/u$	$1/u \times 10^{-1}$	1	10	10^2	10^3	10^4	10^5	10^6
1.0	[1]0.00000	0.21938	1.82292	4.03793	6.33154	8.63322	10.93572	13.23830
1.2	.00003	.29255	1.98932	4.21859	6.51369	8.81553	11.11804	13.42062
1.5	.00017	.39841	2.19641	4.44007	6.73667	9.03866	11.34118	13.64376
2.0	.00115	.55977	2.46790	4.72610	7.02419	9.32632	11.62886	13.93144
2.5	.00378	.70238	2.68126	4.94824	7.24723	9.54945	11.85201	14.15459
3.0	.00857	.82889	2.85704	5.12990	7.42949	9.73177	12.03433	14.33691
3.5	.01566	.94208	3.00650	5.28357	7.58359	9.88592	12.18847	14.49106
4.0	.02491	1.04428	3.13651	5.41675	7.71708	10.01944	12.32201	14.62459
5.0	.04890	1.22265	3.35471	5.63939	7.94018	10.24258	12.54515	14.84773
6.0	.07833	1.37451	3.53372	5.82138	8.12247	10.42490	12.72747	15.03006
7.0	.11131	1.50661	3.68551	5.97529	8.27659	10.57905	12.88162	15.18421
8.0	.14641	1.62342	3.81727	6.10865	8.41011	10.71258	13.01515	15.31774
9.0	.18266	1.72811	3.93367	6.22629	8.52787	10.83036	13.13294	15.43551

$1/u$	$1/u \times 10^7$	10^8	10^9	10^{10}	10^{11}	10^{12}	10^{13}	10^{14}
1.0	15.54087	17.84344	20.14604	22.44862	24.75121	27.05379	29.35638	31.65897
1.2	15.72320	18.02577	20.32835	22.63094	24.93353	27.23611	29.53870	31.84128
1.5	15.94634	18.24892	20.55150	22.85408	25.15668	27.45926	29.76184	32.06442
2.0	16.23401	18.53659	20.83919	23.14177	25.44435	27.74693	30.04953	32.35211
2.5	16.45715	18.75974	21.06233	23.36491	25.66750	27.97008	30.27267	32.57526
3.0	16.63948	18.94206	21.24464	23.54723	25.84982	28.15240	30.45499	32.75757
3.5	16.79362	19.09621	21.39880	23.70139	26.00397	28.30655	30.60915	32.91173
4.0	16.92715	19.22975	21.53233	23.83492	26.13750	28.44008	30.74268	33.04526
5.0	17.15030	19.45288	21.75548	24.05806	26.36064	28.66322	30.96582	33.26840
6.0	17.33263	19.63521	21.93779	24.24039	26.54297	28.84555	31.14813	33.45071
7.0	17.48677	19.78937	22.09195	24.39453	26.69711	28.99969	31.30229	33.60487
8.0	17.62030	19.92290	22.22548	24.52806	26.83064	29.13324	31.43582	33.73840
9.0	17.73808	20.04068	22.34326	24.64584	26.94843	29.25102	31.55360	33.85619

[1]Value shown as 0.00000 is nonzero but less than 0.000005.

Solution 2: Constant discharge from a partially penetrating well in a nonleaky aquifer

Assumptions:

1. Well discharges at a constant rate, Q.
2. Well is of infinitesimal diameter and is screened in only part of the aquifer.
3. Aquifer has radial-vertical anisotropy.
4. Aquifer is not leaky.
5. Discharge from the well is derived exclusively from storage in the aquifer.

Differential equation:

$$\frac{\partial^2 s}{\partial r^2} + \frac{1}{r}\frac{\partial s}{\partial r} + a^2\frac{\partial^2 s}{\partial z^2} = \frac{S}{T}\frac{\partial s}{\partial t}$$

$$a^2 = K_z/K_r$$

This is the differential equation for nonsteady radial and vertical flow in a homogeneous confined aquifer with radial-vertical anisotropy.

Boundary and initial conditions:

$$s(r, z, 0) = 0,\ r \geq 0,\ 0 \leq z \leq b \qquad (1)$$

$$s(\infty, z, t) = 0,\ t \geq 0 \qquad (2)$$

$$\partial s(r, 0, t)/\partial z = 0,\ r \geq 0,\ t \geq 0 \qquad (3)$$

$$\partial s(r, b, t)/\partial z = 0,\ r \geq 0,\ t \geq 0 \qquad (4)$$

$$\lim_{r \to 0} r\frac{\partial s}{\partial r} = \begin{cases} 0, & 0 < z < d \\ -Q/(2\pi K_r(l-d)), & d < z < l \\ 0, & l < z < b \end{cases} \qquad (5)$$

Equation 1 states that initially the drawdown is zero everywhere in the aquifer. Equation 2 states that the drawdown approaches zero as the distance from the pumped well approaches infinity. Equations 3 and 4 state that there is no vertical flow at the upper and lower boundaries of the aquifer. This means that vertical head gradients in the aquifer are caused by the geometric placement of the pumping well screen, and not by leakage. Equation 5 states that near the pumping well the flow is radial, that the flow toward the well is equal to its discharge, that the discharge is distributed uniformly over the well screen, and that no radial flow occurs above and below the screen.

Solution:

I. For the drawdown in a piezometer, a solution by Hantush (1961a, p. 85, and 1964a, p. 353) is given by

$$s = \frac{Q}{4\pi T}\left[W(u) + f\left(u, \frac{ar}{b}, \frac{l}{b}, \frac{d}{b}, \frac{z}{b}\right)\right], \qquad (6)$$

where

$$W(u) = \int_u^\infty \frac{e^{-y}}{y}\,dy$$

and

$$f\left(u, \frac{ar}{b}, \frac{l}{b}, \frac{d}{b}, \frac{z}{b}\right) = \frac{2b}{\pi(l-d)}\sum_{n=1}^{\infty}\frac{1}{n}\left(\sin\frac{n\pi l}{b} - \sin\frac{n\pi d}{b}\right)\cos\frac{n\pi z}{b}\ W\left(u, \frac{n\pi ar}{b}\right) \qquad (7)$$

$$W(u, x) = \int_u^\infty (\exp(-y - x^2/4y)/y)\,dy$$

$$u = \frac{r^2 S}{4Tt}$$

$$a = \sqrt{K_z/K_r}$$

An alternate form of this solution for $a = 1$ is given by Hantush (1961a, p. 85):

$$s = \frac{Qb}{8\pi T(l-d)}\left[M\left(u, \frac{l+z}{r}\right) + M\left(u, \frac{l-z}{r}\right) + f'\left(u, \frac{b}{r}, \frac{l}{r}, \frac{z}{r}\right) - M\left(u, \frac{d+z}{r}\right) - M\left(u, \frac{d-z}{r}\right) - f'\left(u, \frac{b}{r}, \frac{d}{r}, \frac{z}{r}\right)\right], \qquad (8)$$

in which

$$f'\left(u, \frac{b}{r}, \frac{x}{r}, \frac{z}{r}\right) = \sum_{1}^{\infty}\left[M\left(u, \frac{2nb+x+z}{r}\right) - M\left(u, \frac{2nb-x-z}{r}\right) + M\left(u, \frac{2nb+x-z}{r}\right) - M\left(u, \frac{2nb-x+z}{r}\right)\right] \qquad (9)$$

and

$$M(u,\beta)=\int_u^x \frac{e^{-y}}{y}\, erf(\beta\sqrt{y})\, dy$$

$$erf(x)=\frac{2}{\sqrt{\pi}}\int_0^x e^{-y^2}\, dy.$$

II. For the drawdown in an observation well (Hantush, 1961a, p. 90, and 1964a, p. 353),

$$s=\frac{Q}{4\pi T}\left[W(u)+\bar{f}\left(u,\frac{ar}{b},\frac{l}{b},\frac{d}{b},\frac{l'}{b},\frac{d'}{b}\right)\right], \quad (10)$$

where $W(u)$ is as defined previously and

$$\bar{f}\left(u,\frac{ar}{b},\frac{l}{b},\frac{d}{b},\frac{l'}{b},\frac{d'}{b}\right)=\frac{2b^2}{\pi^2(l-d)(l'-d')}\cdot\sum_{n=1}^{\infty}\frac{1}{n^2}\left(\sin\frac{n\pi l}{b}-\sin\frac{n\pi d}{b}\right)\cdot\left(\sin\frac{n\pi l'}{b}-\sin\frac{n\pi d'}{b}\right)W\left(u,\frac{n\pi ar}{b}\right), \quad (11)$$

where $W(u,x)$ and u are as defined previously.

Comments:

Assumptions apply to conditions shown in figure 2.1. The effects of partial penetration need to be considered for $ar/b<1.5$. There must be a type curve for each value of ar/b, d/b, l/b, and either z/b for piezometer, or l'/b and d'/b for observation wells. Because the number of possible type curves is large, only samples of curves for selected values of the parameters are shown in figure 2.2 on plate 1.

For large values of time, that is, for $t>b^2S/(2a^2T)$ or $t>bS/(2K_z)$, the effects of partial penetration are constant in time, and

$$W\left(u,\frac{n\pi ar}{b}\right)$$

can be approximated by

$$2K_0\left(\frac{n\pi ar}{b}\right)$$

(Hantush, 1961a, p. 92). $K_0(x)$ is the modified Bessel function of the second kind of order zero.

Equation 6 then becomes

$$s=\frac{Q}{4\pi T}W(u)+\partial s=\frac{Q}{4\pi T}\left[W(u)+f_s\right],$$

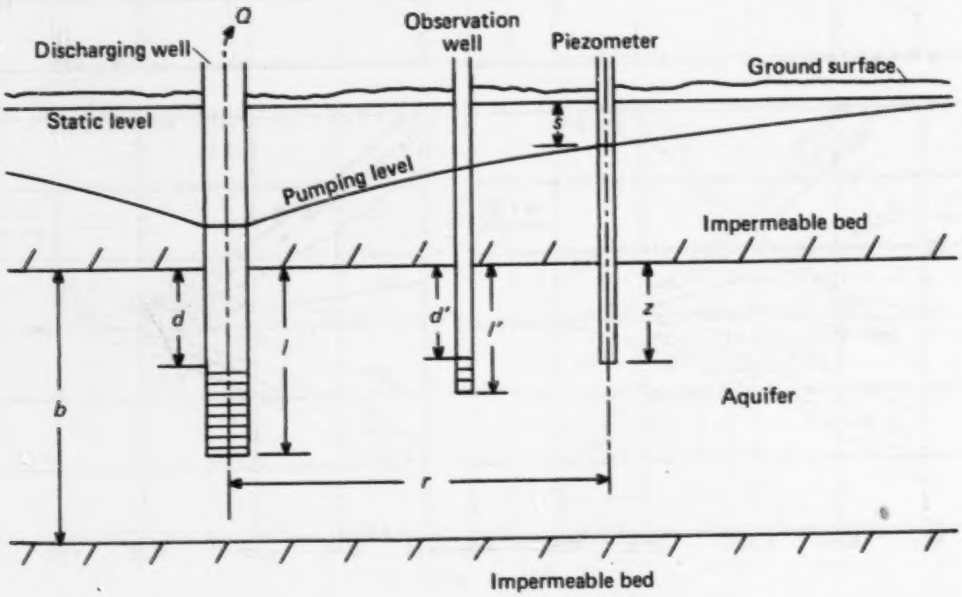

FIGURE 2.1.—Cross section through a discharging well that is screened in a part of a nonleaky aquifer.

where $\partial s = \frac{Q}{4\pi T} f_s$,

and f_s is given in equation 7

with $W\left(u, \frac{n\pi ar}{b}\right)$ replaced by $2K_0\left(\frac{n\pi ar}{b}\right)$.

Figure 2.3 shows plots of f_s as tabulated by Weeks (1969, p. 202–207). In using these curves, it should be noted that f_s for a given r, b, and z_1, l_1, d_1 is equal to f_s for the same r, b, and $z_2=b-z_1$, $l_2=b-d_1$, and $d_2=b-l_1$. Figure 2.3 can be used to find f_s by interpolation and then constructing type curves of $W(u)+f_s$ in the manner described by Weeks (1964, p. D195).

For small values of time

$$t < \frac{(2b-l-z)^2 S}{20T}$$

(Hantush, 1961b, p. 172), equation 8 can be approximated by

$$s = \frac{Qb}{8\pi T(l-d)}\left[M\left(u, \frac{l+z}{r}\right) - M\left(u, \frac{d+z}{r}\right) + M\left(u, \frac{l-z}{r}\right) - M\left(u, \frac{d-z}{r}\right)\right].$$

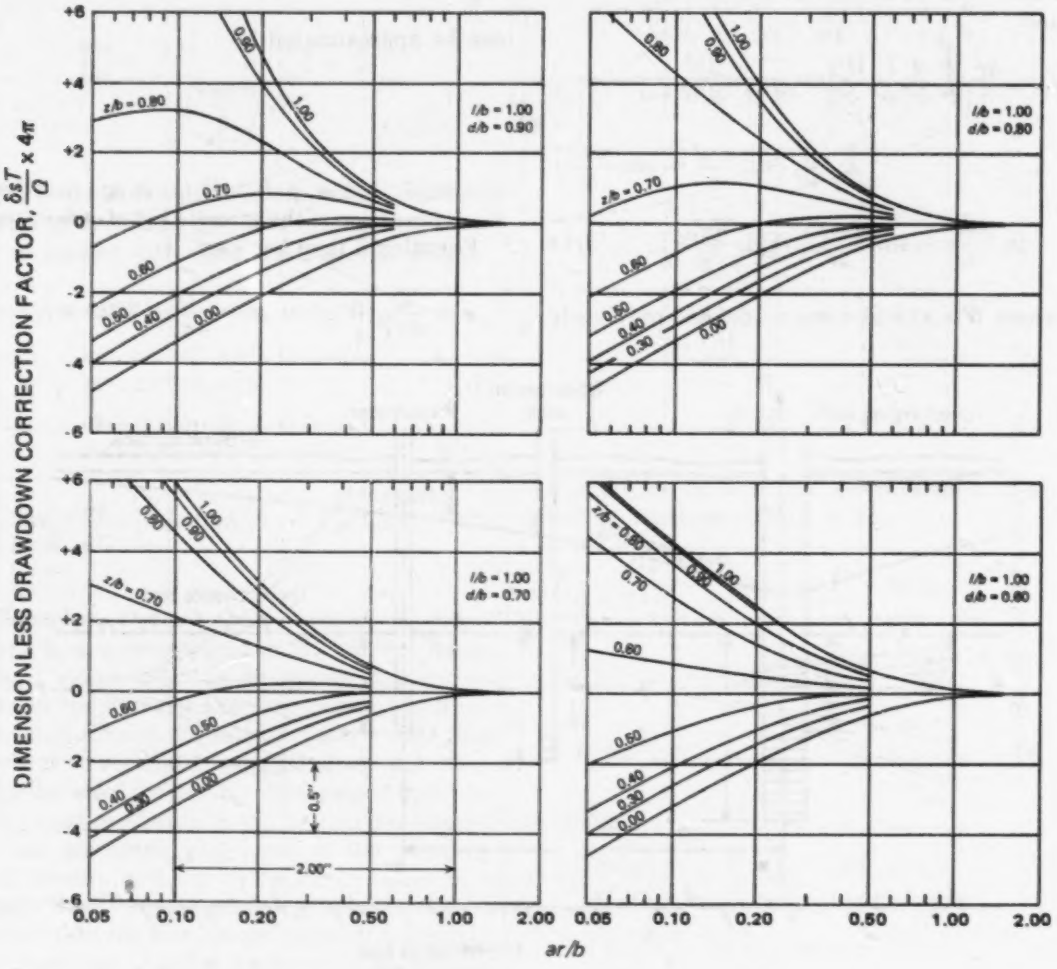

FIGURE 2.3.—The drawdown correction factor f_s versus ar/b, from tables of Weeks (1969).

An extensive table of $M(u,\beta)$ has been prepared by Hantush (1961c).

Although r/b for a given observation well probably would be known, however, the conductivity ratio a^2 would not be. Thus, it would not be known which ar/b curve should be matched. In other words, not only T and S, but also the conductivity ratio a^2 must be determined. A criterion for determining the match between data curves and type curves is that the values of ar/b for different observation wells should all indicate the same "a". Plotting the drawdown data for several observation wells on a single t/r^2 plot and matching to sets of type curves, a different set for each "a", is a useful approach.

Figure 2.2 was prepared from data calculated by the FORTRAN program listed in table 2.1. This program computes "s" from either equation 6 or 10, depending on the input data. The input data consist of cards containing the parameters coded in specific formats. Readers unfamiliar with FORTRAN format items should consult a FORTRAN language manual. The first card contains: the aquifer thickness (b), coded in columns 1–5, in format F5.1; the depth to bottom of pumped well screen (l), coded in columns 6–10, in format F5.1; the

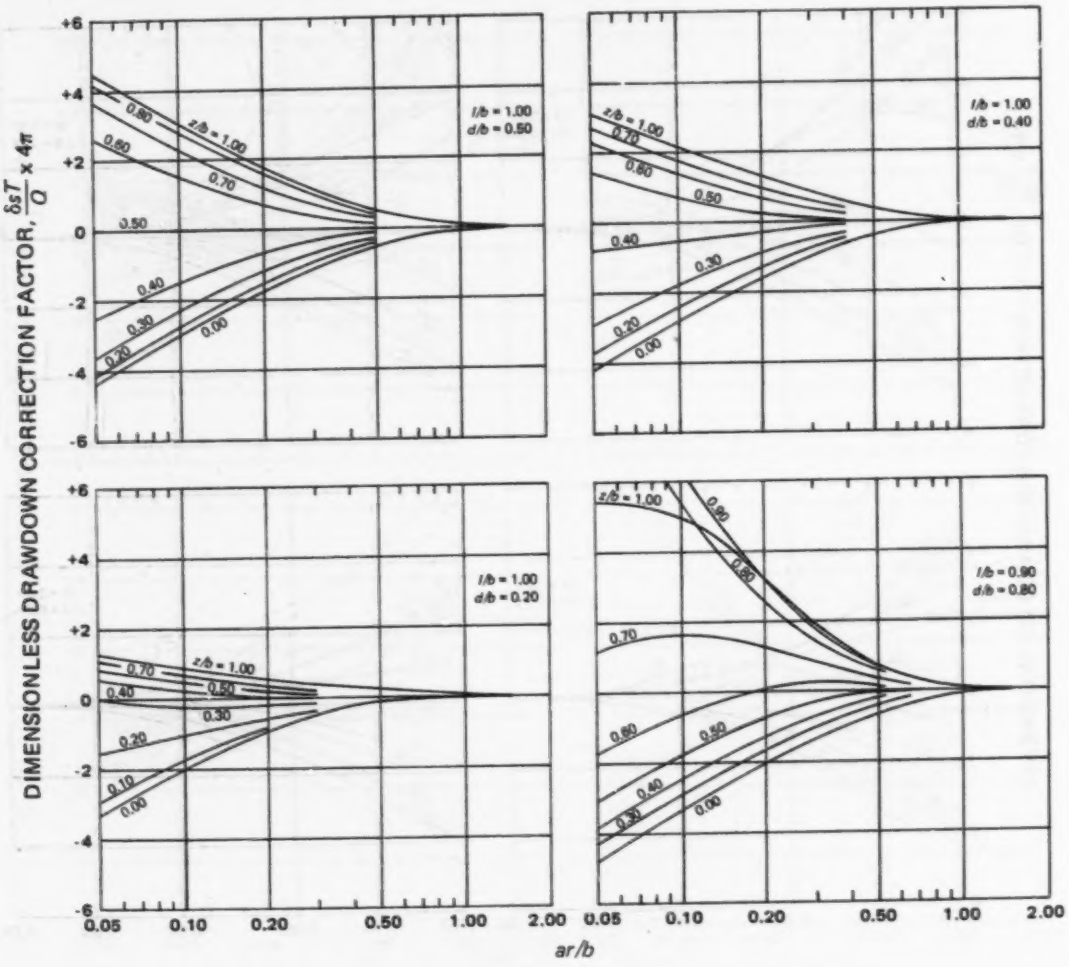

FIGURE 2.3.—Continued.

depth to top of pumped well screen (d), coded in columns 11–15, in format F5.1; the number of observation wells and (or) piezometers, coded in columns 16–20, in format I5; the smallest value of $1/u$ for which computation is desired, coded in columns 21–30, in format E10.4; the largest value of $1/u$ for which computation is desired, coded in columns 31–40, in format E10.4. The ratio of the largest $1/u$ value to the smallest $1/u$ value should be less than 10^{12}. Following this card is a group of cards containing one card for each observation well or piezometer. These cards are coded for an observation well as: distance from pumped well multiplied by the square root of the ratio of the vertical to horizontal conductivity ($r\sqrt{K_z/K_r}$), in columns 1–5, in format F5.1; depth to bottom of observation well screen (l'), coded in columns 6–10, in format F5.1; depth to top of observation well screen (d'), coded in columns 11–15, in format F5.1. A card would be coded for a piezometer as follows: distance from pumped well multiplied by the square root of the ratio of the vertical to horizontal conductivity ($r\sqrt{K_z/K_r}$), in columns 1–5, in format F5.1; and total depth of piezometer (z), in columns 11–15, in format F5.1. The output from this program is tables of computed function values,

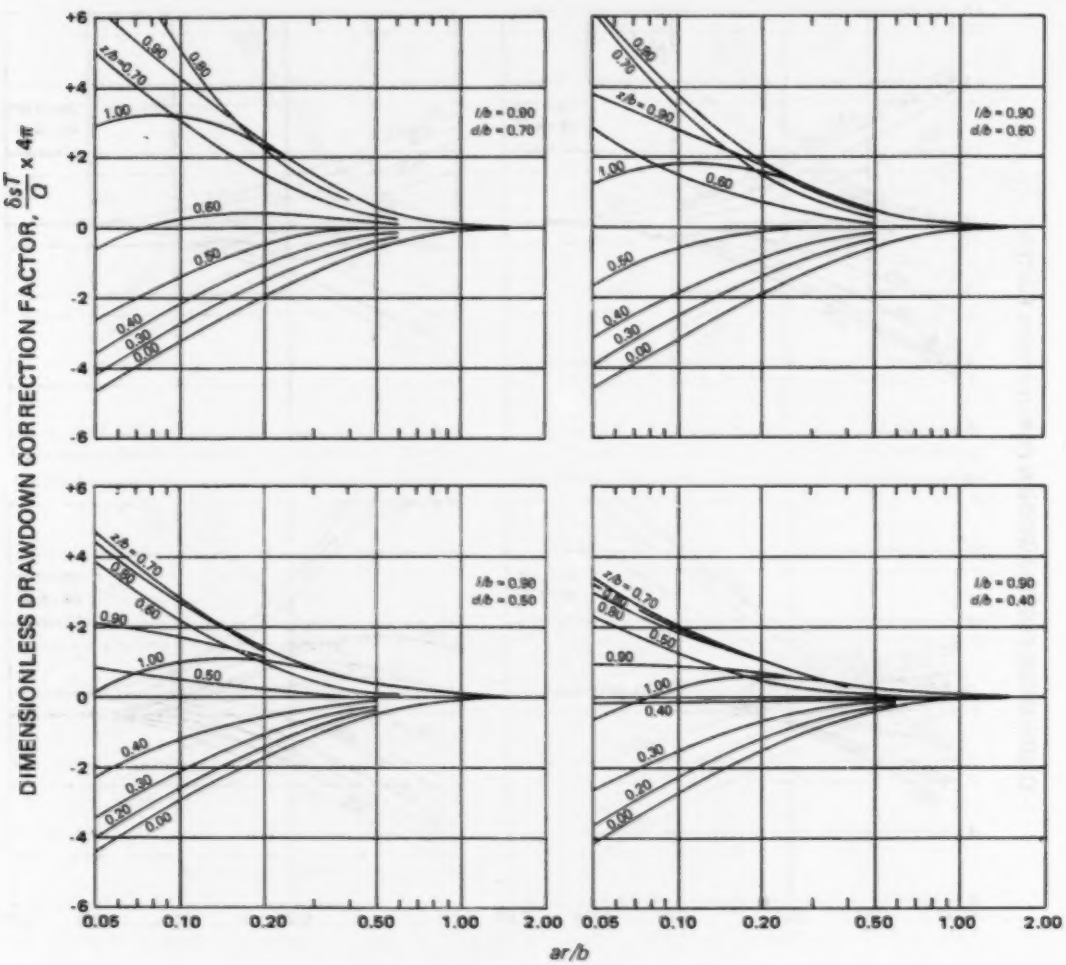

FIGURE 2.3.—Continued.

an example of which is shown in figure 2.4. Subroutines DQL12, BESK, and EXPI are from the IBM Scientific Subroutine Package and a discussion of them is in the IBM SSP manual.

Solution 3: Constant drawdown in a well in a nonleaky aquifer

Assumptions:

1. Water level in well is changed instantaneously by s_w at $t = 0$.
2. Well is of finite diameter and fully penetrates the aquifer.
3. Aquifer is not leaky.
4. Discharge from the well is derived exclusively from storage in the aquifer.

Differential equation:

$$\frac{\partial^2 s}{\partial r^2} + \frac{1}{r}\frac{\partial s}{\partial r} = \frac{S}{T}\frac{\partial s}{\partial t}$$

This is the differential equation describing nonsteady radial flow in a homogeneous isotropic confined aquifer.

Boundary and initial conditions:

$$s(r,0) = 0, r \geq r_w \quad (1)$$

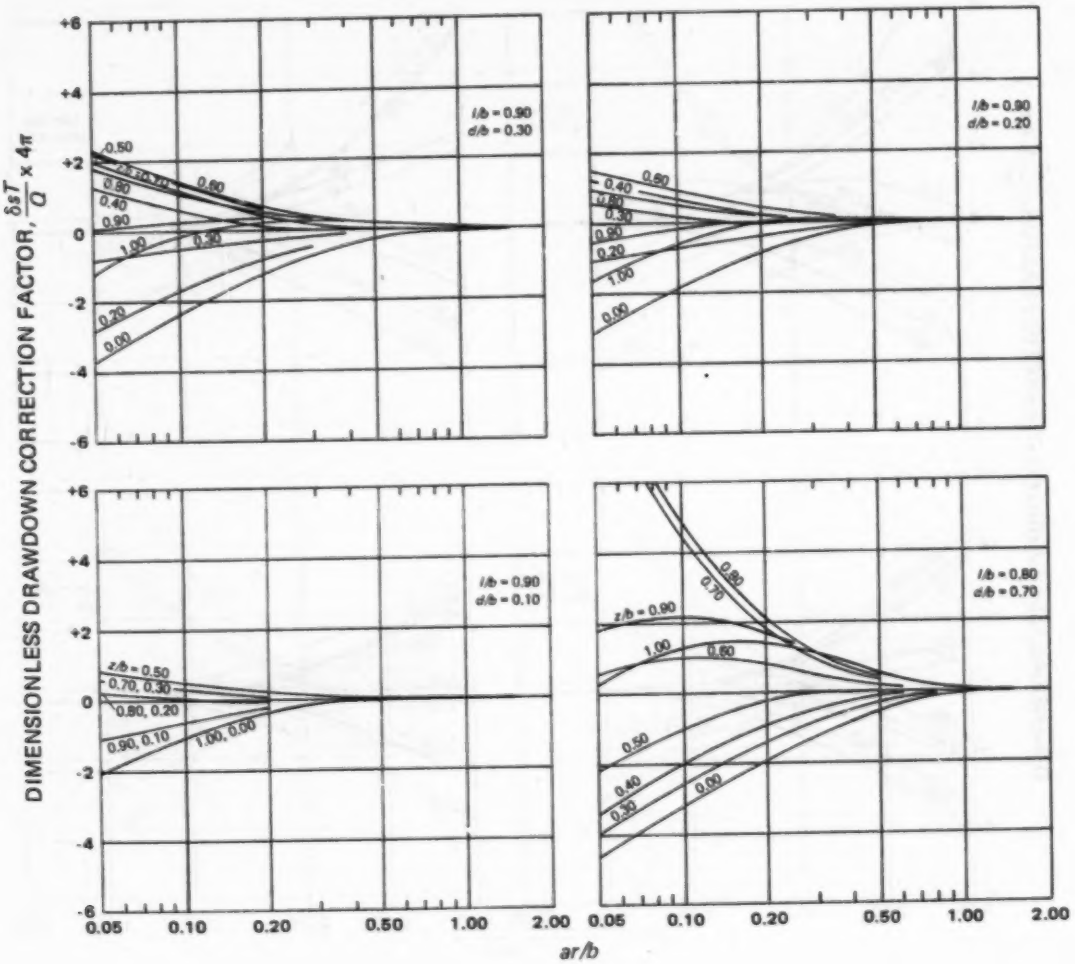

FIGURE 2.3.—Continued.

$$s(r_w,t) = \begin{cases} 0, t < 0 \\ s_w = \text{constant}, t \geq 0 \end{cases} \quad (2)$$

$$s(\infty,t) = 0, t \geq 0 \quad (3)$$

Equation 1 states that initially the drawdown is zero everywhere in the aquifer. Equation 2 states that, as the well is approached, drawdown in the aquifer approaches the constant drawdown in the well, implying no entrance loss to the well. Equation 3 states that the drawdown approaches zero as the distance from the well approaches infinity.

Solutions:

I. For the well discharge (Jacob and Lohman, 1952, p. 560):

$$Q = 2\pi T\, s_w\, G(\alpha),$$

where

$$G(\alpha) = \frac{4\alpha}{\pi}\int_0^\infty x e^{-\alpha x^2}\left\{\frac{\pi}{2} + \tan^{-1}\left[\frac{Y_0(x)}{J_0(x)}\right]\right\} dx$$

and

$$\alpha = \frac{Tt}{Sr_w^2}.$$

II. For the drawdown in water level (Hantush, 1964a, p. 343):

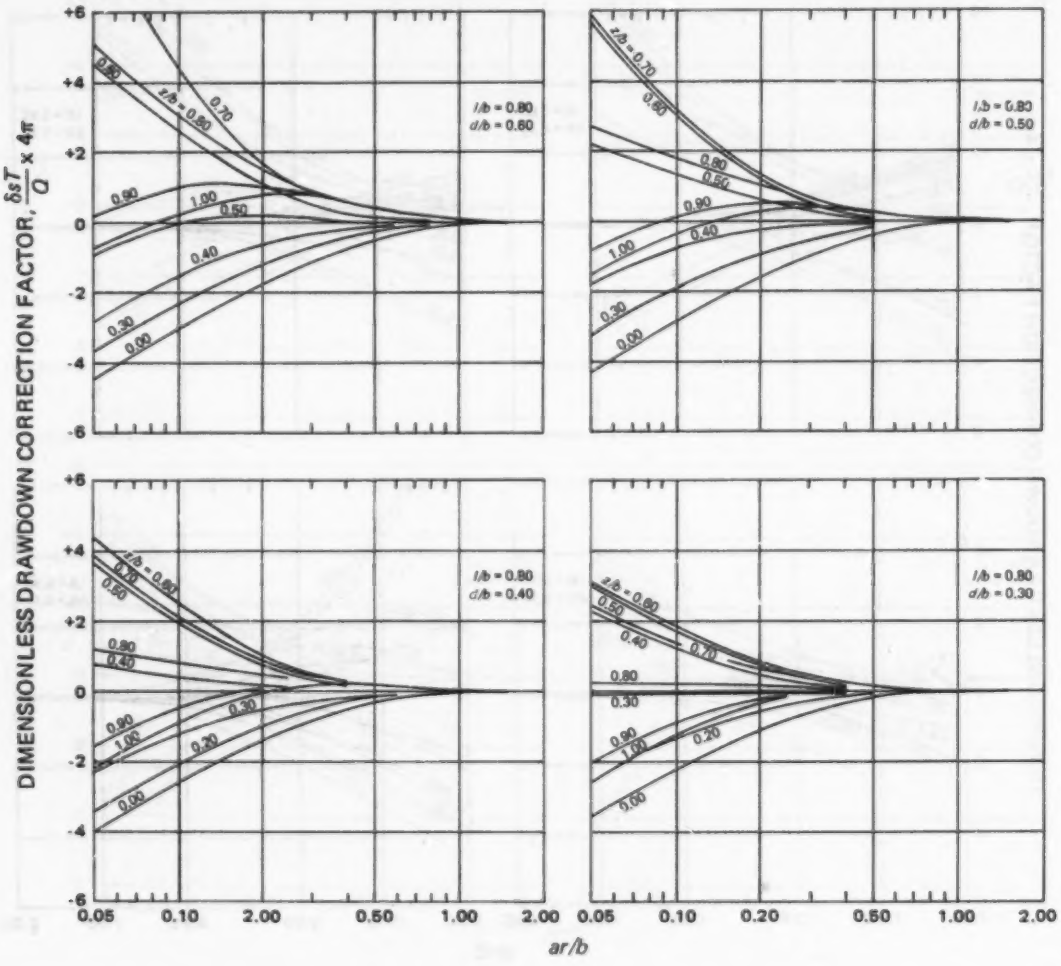

FIGURE 2.3.—Continued.

$$s = s_{ir}\, A(\tau, \rho),$$

where $A(\tau,\rho) = 1$

$$-\frac{2}{\pi}\int_0^\infty \frac{J_0(u)\, Y_0(\rho u) - Y_0(u) J_0(\rho u)}{J_0^2(u) + Y_0^2(u)} \exp(-\tau u^2)\frac{du}{u},$$

and

$$\tau = \alpha = \frac{Tt}{Sr_{ir}^2},$$

$$\rho = \frac{r}{r_{ir}}.$$

Comments:

Boundary condition 2 requires a constant drawdown in the discharging well, a condition most commonly fulfilled by a flowing well, although figure 3.1 shows the water level to be below land surface.

Figure 3.2 on plate 1 is a plot from Lohman (1972, p. 24) of dimensionless discharge ($G(\alpha)$) versus dimensionless time (α). Additional values in the range α greater than 1×10^{12} were calculated from $G(\alpha)\simeq 2/\log(2.2458\alpha)$ (Hantush, 1964a, p. 312). Function values for $G(\alpha)$ are given in table 3.1. The data curve consists of measured well discharge versus time. After the data and type curves are matched, transmissivity can be calculated from $T = Q/2\pi s_{ir} G(\alpha)$, and the storage coefficient can be

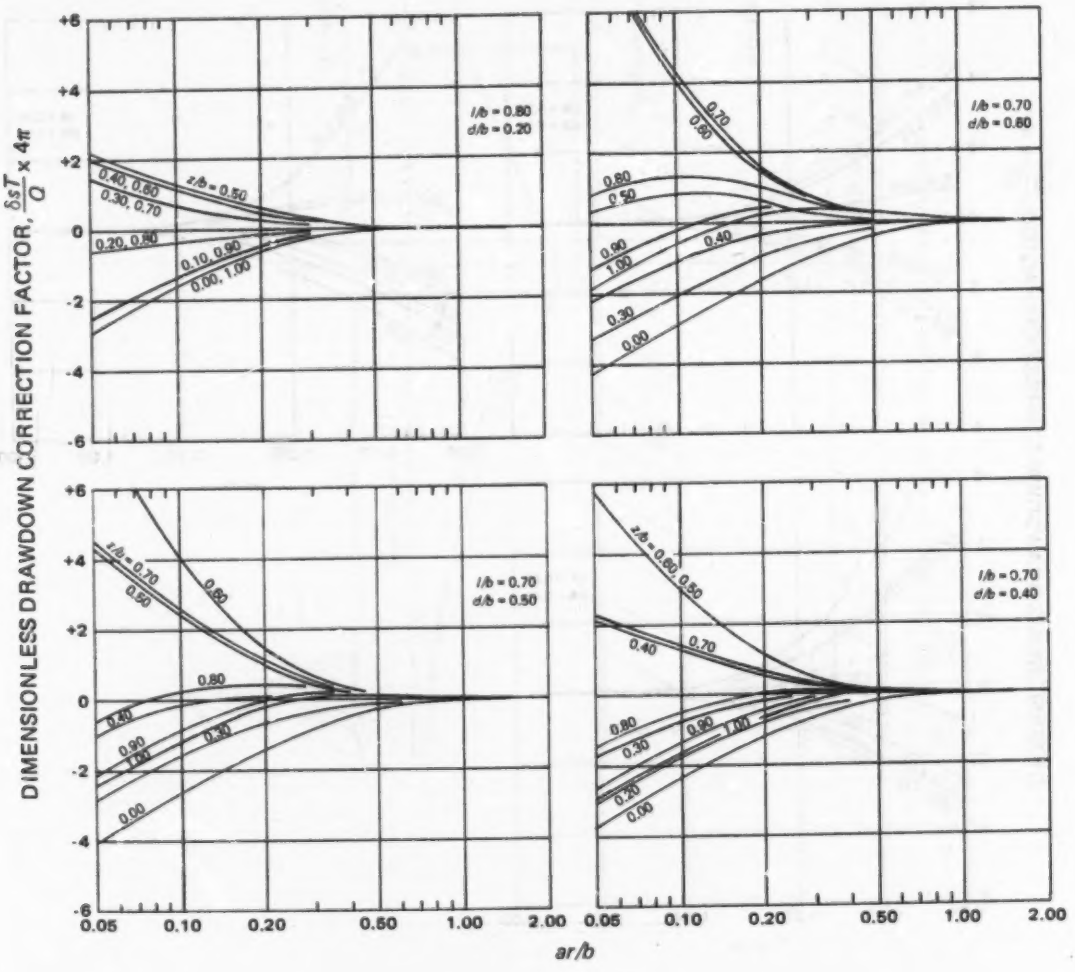

FIGURE 2.3.—Continued.

calculated from $S = Tt/\alpha r_w^2$, where $(\alpha, G(\alpha))$ and (t, Q) are matching points on the type curve and data curve, respectively.

Similarly, data curves of drawdown versus time may be matched to figure 3.3 on plate 1; this is a plot of dimensionless drawdown $(A(\tau,\rho) = s/s_w)$ versus dimensionless time $(\tau/\rho^2 = Tt/Sr^2)$. After the data and type curves are matched, the hydraulic diffusivity of the aquifer can be calculated from the equality $T/S = (\tau/\rho^2)(r^2/t)$. Usually s_w is known, and some of the uncertainty of curve matching can be eliminated by plotting s/s_w versus t because only horizontal translation is then required. If r_w is also known, the particular curve to be matched can be determined from the relation $\rho = r/r_w$. Generally, however, the effective radius, r_w, differs from the actual radius and is not known. The effective radius can often be estimated from a knowledge of the construction of the well and the water-bearing material, or it can be determined from step-drawdown tests (Rorabaugh, 1953). Figure 3.3 was plotted from table 3.2. For $\tau \leq 1\times10^3$, the data are from Hantush (1964a, p. 310). For $\tau > 1\times10^3$, values of drawdown in a leaky aquifer, as $r_w/B \rightarrow 0$, were used. (See solution 7.) Where 0.000 occurs in table 3.2, $A(\tau,\rho)$ is less than 0.0005.

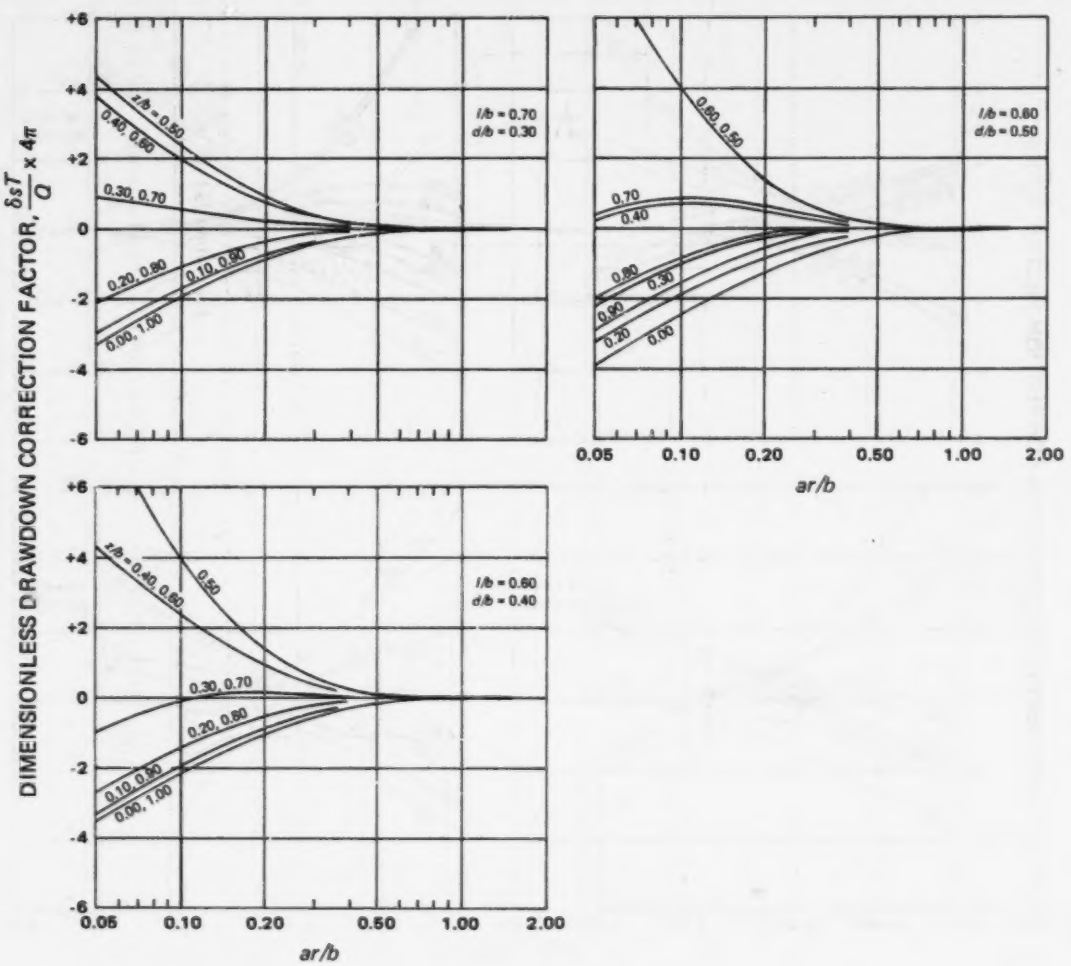

FIGURE 2.3.—Continued.

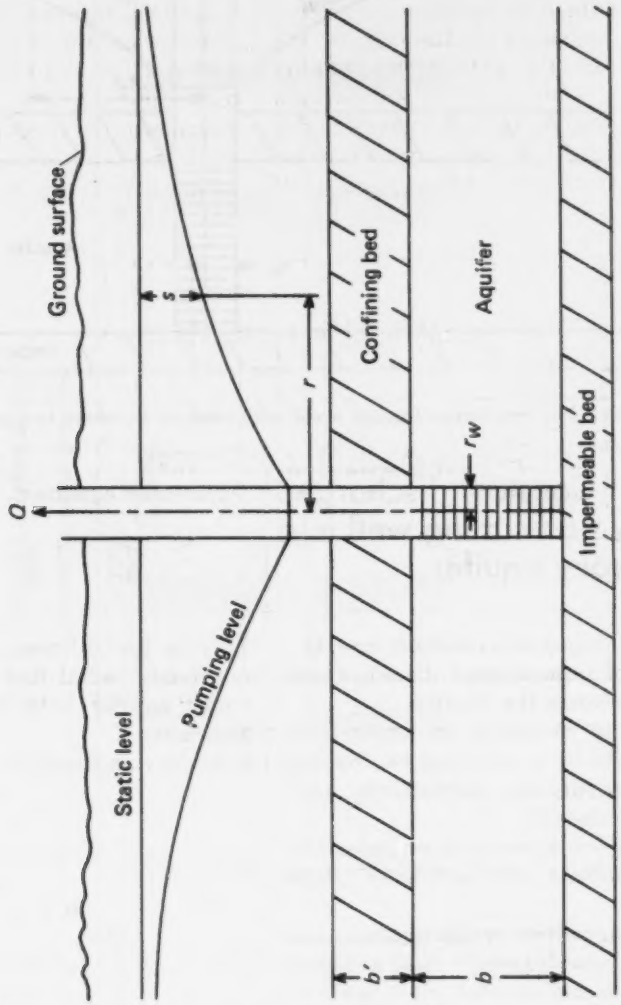

FIGURE 2.4.—Example of output from program for partial penetration in a nonleaky artesian aquifer.

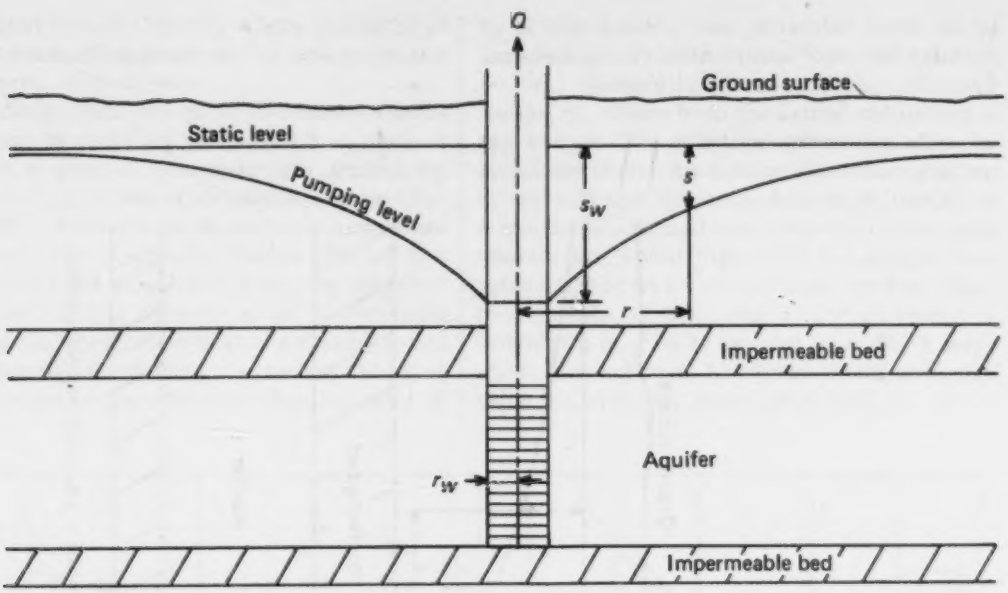

FIGURE 3.1.—Cross section through a well with constant drawdown in a nonleaky aquifer.

Solution 4: Constant discharge from a fully penetrating well in a leaky aquifer

Assumptions:

1. Well discharges at a constant rate, Q.
2. Well is of infinitesimal diameter and fully penetrates the aquifer.
3. Aquifer is overlain, or underlain, everywhere by a confining bed having uniform hydraulic conductivity (K') and thickness (b').
4. Confining bed is overlain, or underlain, by an infinite constant-head plane source.
5. Hydraulic gradient across confining bed changes instantaneously with a change in head in the aquifer (no release of water from storage in the confining bed).
6. Flow in the aquifer is two-dimensional and radial in the horizontal plane and flow in the confining bed is vertical. This assumption is approximated closely where the hydraulic conductivity of the aquifer is sufficiently greater than that of the confining bed.

Differential equation:

$$\frac{\partial^2 s}{\partial r^2} + \frac{1}{r}\frac{\partial s}{\partial r} - \frac{sK'}{Tb'} = \frac{S}{T}\frac{\partial s}{\partial t}$$

This is the differential equation describing nonsteady radial flow in a homogeneous isotropic aquifer with leakage proportional to drawdown.

Boundary and initial conditions:

$$s(r,0)=0, \quad r \geq 0 \tag{1}$$

$$s(\infty,t)=0, \quad t \geq 0 \tag{2}$$

$$Q = \begin{cases} 0, & t<0 \\ \text{constant}>0, & t \geq 0 \end{cases} \tag{3}$$

$$\lim_{r \to 0} r \frac{\partial s}{\partial r} = -\frac{Q}{2\pi T} \tag{4}$$

Equation 1 states that the initial drawdown is zero. Equation 2 states that drawdown is small at a large distance from the pumping well. Equation 3 states that the discharge from the well is constant and begins at $t=0$. Equation 4 states that near the pumping well the flow toward the well is equal to its discharge.

TABLE 3.1.—*Values of* $G(\alpha)$

[Modified from Lohman (1972, p. 24)]

α	$\alpha \times 10^{-4}$	10^{-3}	10^{-2}	10^{-1}	1	10	10^{2}	10^{3}	10^{4}	10^{5}
1	56.9	18.34	6.13	2.249	0.985	0.534	0.346	0.251	0.1964	0.1608
2	40.4	13.11	4.47	1.716	.803	.461	.311	.232	.1841	.1524
3	33.1	10.79	3.74	1.477	.719	.427	.294	.222	.1777	.1479
4	28.7	9.41	3.30	1.333	.667	.405	.283	.215	.1733	.1449
5	25.7	8.47	3.00	1.234	.630	.389	.274	.210	.1701	.1426
6	23.5	7.77	2.78	1.160	.602	.377	.268	.206	.1675	.1408
7	21.8	7.23	2.60	1.103	.580	.367	.263	.203	.1654	.1393
8	20.4	6.79	2.46	1.057	.562	.359	.258	.200	.1636	.1380
9	19.3	6.43	2.35	1.018	.547	.382	.254	.198	.1621	.1369

α	$\alpha \times 10^{6}$	10^{7}	10^{8}	10^{9}	10^{10}	10^{11}	10^{12}	10^{13}	10^{14}	10^{15}
1	0.1360	0.1177	0.1037	0.0927	0.0838	0.0764	0.0704	0.0651	0.0605	0.0566
2	.1299	.1131	.1002	.0899	.0814	.0744	.0686	.0636	.0593	.0555
3	.1266	.1106	.0982	.0883	.0801	.0733	.0677	.0628	.0586	.0549
4	.1244	.1089	.0968	.0872	.0792	.0726	.0671	.0622	.0581	
5	.1227	.1076	.0958	.0864	.0785	.0720	.0666	.0618	.0577	
6	.1213	.1066	.0950	.0857	.0779	.0716	.0662	.0615	.0574	
7	.1202	.1057	.0943	.0851	.0774	.0712	.0658	.0612	.0572	
8	.1192	.1049	.0937	.0846	.0770	.0709	.0655	.0609	.0569	
9	.1184	.1043	.0932	.0842	.0767	.0706	.0653	.0607	.0567	

TABLE 3.2.—*Values of* $A(\tau,\rho)$

[Values of $A(\tau,\rho)$ for $\tau \leq 10^2$ modified from Hantush (1964a, p. 310)]

τ		ρ							
		5	10	20	50	100	200	500	1000
1	× 1	0.002							
2		.022							
3		.049							
4		.076	0.000						
5		.101	.002						
7		.142	.006						
1	× 10	.188	.016	0.000					
2		.277	.057	.001					
3		.325	.094	.004					
4		.358	.123	.009					
5		.381	.146	.016					
7		.414	.184	.031					
1	× 10^2	.446	.222	.053	0.000				
1.5		.479	.264	.085	.001				
2		.500	.291	.110	.003				
3		.528	.328	.146	.009				
5		.559	.372	.194	.026	0.000			
7		.578	.397	.223	.044	.001			
1	× 10^3	.596	.422	.254	.066	.004			
1.5		.615	.450	.287	.094	.012			
2		.627	.467	.309	.116	.021	0.000		
3		.644	.490	.338	.147	.039	.001		
5		.662	.517	.372	.186	.068	.006		
7		.673	.533	.392	.211	.089	.014		
1	× 10^4	.685	.549	.413	.237	.114	.025		
1.5		.696	.566	.435	.264	.142	.043	0.000	
2		.704	.577	.450	.283	.161	.058	.001	
3		.715	.592	.469	.308	.188	.081	.005	
5		.727	.609	.492	.337	.221	.113	.014	0.000
7		.734	.620	.506	.355	.242	.134	.025	.001
1	× 10^5	.742	.631	.520	.373	.263	.156	.039	.002
1.50		.750	.642	.532	.392	.285	.180	.058	.007
2		.755	.650	.544	.405	.300	.197	.072	.013
3		.762	.660	.558	.423	.321	.220	.094	.024
5		.771	.672	.574	.443	.345	.247	.122	.044
7		.776	.680	.584	.456	.360	.264	.141	.059
1	× 10^6	.782	.688	.594	.470	.376	.282	.160	.076
1.5		.788	.696	.604	.484	.392	.301	.181	.096
2		.792	.702	.612	.493	.403	.314	.196	.111
3		.797	.709	.622	.506	.418	.331	.215	.132
5		.803	.718	.633	.521	.436	.352	.240	.157
7		.807	.724	.641	.531	.448	.365	.255	.173
1	× 10^7	.811	.730	.648	.541	.459	.378	.270	.190
1.5		.815	.736	.656	.551	.472	.392	.287	.208
2		.818	.740	.662	.558	.480	.402	.299	.221
3		.822	.746	.669	.568	.492	.415	.314	.238
5		.827	.753	.678	.580	.506	.431	.333	.258
7		.830	.757	.684	.587	.514	.441	.344	.271
1	× 10^8	.833	.762	.690	.595	.523	.452	.357	.285
1.5		.837	.766	.696	.603	.533	.463	.370	.300
2		.839	.770	.701	.609	.540	.470	.379	.310
3		.842	.774	.706	.617	.549	.481	.391	.323
5		.846	.780	.714	.626	.560	.494	.406	.340
7		.849	.783	.718	.632	.567	.502	.415	.350
1	× 10^9	.851	.787	.723	.638	.574	.510	.425	.361
1.5		.854	.791	.728	.645	.582	.519	.435	.372
2		.856	.794	.731	.649	.587	.525	.443	.380
3		.858	.797	.736	.655	.594	.533	.452	.392
5		.861	.802	.742	.663	.603	.544	.464	.405
7		.863	.804	.746	.668	.609	.550	.472	.413
1	× 10^{10}	.865	.807	.749	.673	.615	.557	.480	.422
1.5		.867	.810	.753	.678	.621	.564	.488	.431
2		.869	.813	.756	.682	.625	.569	.494	.438
3		.871	.816	.760	.687	.631	.576	.502	.447
5		.874	.819	.765	.693	.638	.584	.512	.457
7		.875	.821	.768	.696	.643	.589	.518	.464
1	× 10^{11}	.877	.824	.770	.700	.647	.594	.524	.471
1.5									
2									
3									
5									
7									

Solution (Hantush and Jacob, 1955, p. 98):

$$s = \frac{Q}{4\pi T}\int_u^\infty \frac{e^{-z-\frac{r^2}{4B^2z}}}{z} \qquad (5)$$

where $u = r^2S/4Tt$

$$B = \sqrt{\frac{Tb'}{K'}}. \qquad (6)$$

Comments:

As pointed out by Hantush and Jacob (1954, p. 917), leakage is three-dimensional, but if the difference in hydraulic conductivities of the aquifer and confining bed are sufficiently great, the flow may be assumed to be vertical in the confining bed and radial in the aquifer. This relationship has been quantified by Hantush (1967, p. 587) in the condition $b/B<0.1$. In terms of relative conductivities, this would be $K/K' > 100\ b/b'$. Assumption 5, that there is no change in storage of water in the confining bed, was investigated by Neuman and Witherspoon (1969b, p. 821). They concluded that this assumption would not affect the solution if

$$\beta < 0.01, \text{ where } \beta = \frac{r}{4b}\sqrt{\frac{K'S_s'}{KS_s}}.$$

Assumption 4, that there is no drawdown in water level in the source bed lying above the confining bed, was also examined by Neuman and Witherspoon (1969a, p. 810). They indicated that drawdown in the source bed would have negligible effect on drawdown in the pumped aquifer for short times, that is, when $\frac{Tt}{r^2S} < 1.6\frac{\beta^2}{(r/B)^4}$. Also, they indicated (1969a, p. 811) that neglect of drawdown in the source bed is justified if $T_s > 100T$, where T_s represents the transmissivity of the source bed. Figure 4.1, a cross section through the discharging well, shows geometric relationships. Figure 4.2 on plate 1 shows plots of dimensionless drawdown compared to dimensionless time, using the notation of Cooper (1963) from Lohman (1972, pl. 3). Cooper expressed equations 5 and 6 as

$$L(u,v) = \int_u^\infty \frac{e^{-y-\frac{v^2}{y}}}{y}\,dy, \qquad (7)$$

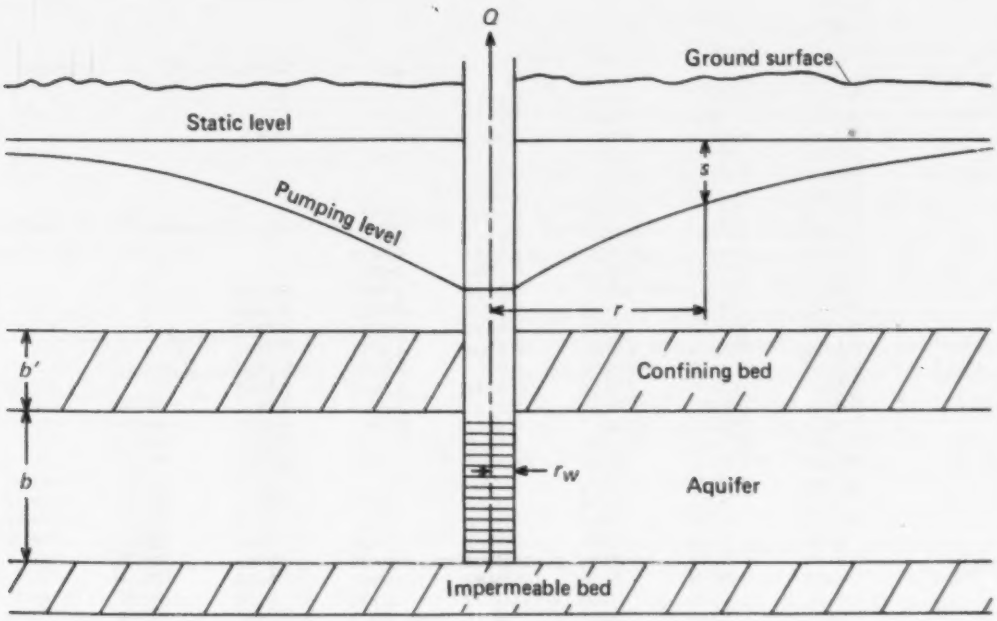

FIGURE 4.1.—Cross section through a discharging well in a leaky aquifer.

with

$$v = \frac{r}{2}\sqrt{\frac{K'}{Tb'}}. \qquad (8)$$

Cooper's type curves and equation 5 express the same function with $r/B=2v$. Hantush (1961e) has a tabulation of equation 5, parts of which are included in table 4.1.

The observed data may be plotted in two ways (Cooper, 1963, p. C51). The measured drawdown in any one well is plotted versus t/r^2; the data are then matched to the solid-line type curves of figure 4.2. The data points are alined with the solid-line type curves either on one of them or between two of them. The parameters are then computed from the coordinates of the match points $(t/r^2,s)$ and $(1/u, L(u,v))$, and an interpolated value of v from the equations

$$T = \frac{Q}{4\pi}\frac{L(u,v)}{s}, \qquad (9)$$

$$S = 4T\frac{t/r^2}{1/u}, \qquad (10)$$

and

$$\frac{K'}{b'} = 4T\frac{v^2}{r^2}.$$

Drawdown measured at the same time but in different observation wells at different distances can be plotted versus t/r^2 and matched to the dashed-line type curves of figure 4.2. The data are matched so as to aline with the dashed-line curves, either on one or between two of them. From the match-point coordinates $(s,t/r^2)$ and $(L(u,v),1/u)$ and an interpolated value of v^2/u, T and S are computed from equations 9 and 10 and the remaining parameter from

$$K'/b' = S\frac{v^2/u}{t}.$$

The region $v^2/u \geq 8$ and $L(u,v) \geq 10^{-2}$ corresponds to steady-state conditions.

TABLE 4.1.—*Selected values of* W(u,r/B)

[From Hantush (1961e)]

u	r/B							
	0.001	0.003	0.01	0.03	0.1	0.3	1	3
1×10^{-6}	13.0031	11.8153	9.4425	7.2471	4.8541	2.7449	0.8420	0.0695
2	12.4240	11.6716						
3	12.0581	11.5098	9.4425					
5	11.5795	11.2248	9.4413					
7	11.2570	10.9951	9.4361					
1×10^{-5}	10.9109	10.7228	9.4176					
2	10.2301	10.1332	9.2961	7.2471				
3	9.8288	9.7635	9.1499	7.2470				
5	9.3213	9.2818	8.8827	7.2450				
7	8.9863	8.9580	8.6625	7.2371				
1×10^{-4}	8.6308	8.6109	8.3983	7.2122				
2	7.9390	7.9290	7.8192	7.0685				
3	7.5340	7.5274	7.4534	6.9068	4.8541			
5	7.0237	7.0197	6.9750	6.6219	4.8530			
7	6.6876	6.6848	6.6527	6.3923	4.8478			
1×10^{-3}	6.3313	6.3293	6.3069	6.1202	4.8292			
2	5.6393	5.6383	5.6271	5.5314	4.7079	2.7449		
3	5.2348	5.2342	5.2267	5.1627	4.5622	2.7448		
5	4.7260	4.7256	4.7212	4.6829	4.2960	2.7428		
7	4.3916	4.3913	4.3882	4.3609	4.0771	2.7350		
1×10^{-2}	4.0379	4.0377	4.0356	4.0167	3.8150	2.7104		
2	3.3547	3.3546	3.3536	3.3444	3.2442	2.5688		
3	2.9591	2.9590	2.9584	2.9523	2.8873	2.4110	.8420	
5	2.4679	2.4679	2.4675	2.4642	2.4271	2.1371	.8409	
7	2.1508	2.1508	2.1506	2.1483	2.1232	1.9206	.8360	
1×10^{-1}	1.8229	1.8229	1.8227	1.8213	1.8050	1.6704	.8190	
2	1.2226	1.2226	1.2226	1.2220	1.2155	1.1602	.7148	.0695
3	.9057	.9057	.9056	.9053	.9018	.8713	.6010	.0694
5	.5598	.5598	.5598	.5596	.5581	.5453	.4210	.0681
7	.3738	.3738	.3738	.3737	.3729	.3663	.2996	.0639
1×10^{0}	.2194	.2194	.2194	.2193	.2190	.2161	.1855	.0534
2	.0489	.0489	.0489	.0489	.0488	.0485	.0444	.0210
3	.0130	.0130	.0130	.0130	.0130	.0130	.0122	.0071
5	.0011	.0011	.0011	.0011	.0011	.0011	.0011	.0008
7	.0001	.0001	.0001	.0001	.0001	.0001	.0001	.0001

The drawdown in the steady-state region is given by the equation (Jacob, 1946, eq. 15)

$$s = \frac{Q}{2\pi T} K_0(x),$$

where $K_0(x)$ is the zero-order modified Bessel function of the second kind and

$$x = r\sqrt{\frac{K'}{Tb'}}.$$

Data for steady-state conditions can be analyzed using figure 4.3 on plate 1. The drawdowns are plotted versus r and matched to figure 4.3. After choosing a convenient match point with coordinates (s,r) and $(K_0(x),x)$ the parameters are computed from the equations

$$T = \frac{Q}{2\pi s} K_0(x) \text{ and } \frac{K'}{b'} = \frac{xT}{r^2}.$$

Values of $K_0(x)$ from Hantush (1956) are given in table 4.2.

A FORTRAN program for generating type-curve function values of equation 7 is listed in table 4.3. Using the notation $L(u,v)$ of Cooper (1963), the function is evaluated as follows. For $u \geq 1$,

$$L(u,v) = \int_u^\infty (1/y) \exp(-y - v^2/y)\, dy = \int_u^\infty f(y)\, dy.$$

This integral is transformed into the form

$$\int_0^\infty e^{-r}\left[\exp\left(-u - \frac{v^2}{x+u}\right)\frac{1}{x+u}\right] dx$$

evaluated by a Gaussian-Laguerre quadrature formula. For $v^2 < u < 1$,

TABLE 4.2.—*Selected values of $K_0(x)$*

[From Hantush (1956, p. 704)]

N	$x = NX10^{-2}$	$x = NX10^{-1}$	$x = N$
1	4.7212	2.4271	0.4210
1.5	4.3159	2.0300	.2138
2	4.0285	1.7527	.1139
3	3.6235	1.3725	.0347
4	3.3365	1.1145	.0112
5	3.1142	.9244	.0037
6	2.9329	.7775	------
7	2.7798	.6605	------
8	2.6475	.5653	------
9	2.5310	.4867	------

$$L(u,v) = \int_1^\infty f(y)\, dy + \int_u^1 f(y)\, dy.$$

The first integral is evaluated by a Gaussian-Laguerre quadrature formula, as previously described. The second integral is evaluated using a series expansion, as

$$\int_u^1 f(y)dy = s(1) - s(u),$$

where

$$s = \log u \left[\sum_{n=0}^{\infty} \frac{(v^2)^n}{(n!)^2}\right] + \sum_{m=1}^{\infty}\left[\frac{(-1)^m}{m}\left[u^m - \left(\frac{v^2}{u}\right)^m\right]\left[\sum_{n=0}^{\infty}\frac{(v^2)^n}{(m+n)!n!}\right]\right].$$

For $u < 1$ and $u \leq v^2$,

$$L(u,v) = 2K_0(2v) - \int_{\frac{v^2}{u}}^\infty f(y)\, dy$$

(Cooper, 1963, p. C50),

where K_0 is the zero-order modified Bessel function of the second kind. The integral in the above expression is evaluated by the Gaussian-Laguerre procedure, as described previously.

Input data for this program consist of three cards with the numeric data coded by specific FORTRAN formats. Readers unfamiliar with FORTRAN format items should consult a FORTRAN language manual. The first card contains: the smallest value of $1/u$ for which computation is desired, coded in columns 1–10 in format E10.5; the largest value of $1/u$ for which computation is desired, coded in columns 11–20 in format E10.5. The table will include a range of $1/u$ values spanning these two coded values if the span is less than or equal to 12 log cycles. The next two cards contain 12 values of r/B, all coded in format E10.5, in columns 1–10, 11–20, 21–30, 31–40, 41–50, 51–60, 61–70, and 71–80 of the first card and columns 1–10, 11–20, 21–30, and 31–40 of the second card. Zero (or blank) coding is permissible in this field, but computation will terminate with the first zero (or blank) value encountered. An example of the output from this program is shown in figure 4.4.

W(U,R/B)

1/U	R/B 0.10E-05	0.30E-05	0.10E-04	0.30E-04	0.10E-03	0.30E-03	0.10E-02	0.30E-02	0.10E-01
0.100E 01	0.2194	0.2194	0.2194	0.2194	0.2194	0.2194	0.2194	0.2194	0.2194
0.150E 01	0.3984	0.3984	0.3984	0.3984	0.3984	0.3984	0.3984	0.3984	0.3984
0.200E 01	0.5598	0.5598	0.5598	0.5598	0.5598	0.5598	0.5598	0.5598	0.5598
0.300E 01	0.8289	0.8289	0.8289	0.8289	0.8289	0.8289	0.8289	0.8289	0.8289
0.500E 01	1.2226	1.2226	1.2226	1.2226	1.2226	1.2226	1.2226	1.2226	1.2226
0.700E 01	1.5066	1.5066	1.5066	1.5066	1.5066	1.5066	1.5066	1.5066	1.5065
0.100E 02	1.8229	1.8229	1.8229	1.8229	1.8229	1.8229	1.8229	1.8229	1.8227
0.150E 02	2.1964	2.1964	2.1964	2.1964	2.1964	2.1964	2.1964	2.1964	2.1961
0.200E 02	2.4679	2.4679	2.4679	2.4679	2.4679	2.4679	2.4679	2.4679	2.4675
0.300E 02	2.8570	2.8570	2.8570	2.8570	2.8570	2.8570	2.8570	2.8570	2.8564
0.500E 02	3.3547	3.3547	3.3547	3.3547	3.3547	3.3547	3.3547	3.3546	3.3536
0.700E 02	3.6855	3.6855	3.6855	3.6855	3.6855	3.6855	3.6855	3.6854	3.6839
0.100E 03	4.0379	4.0379	4.0379	4.0379	4.0379	4.0379	4.0379	4.0377	4.0356
0.150E 03	4.4401	4.4401	4.4401	4.4401	4.4401	4.4401	4.4400	4.4397	4.4365
0.200E 03	4.7261	4.7261	4.7261	4.7261	4.7261	4.7261	4.7260	4.7257	4.7212
0.300E 03	5.1299	5.1299	5.1299	5.1299	5.1299	5.1299	5.1298	5.1292	5.1226
0.500E 03	5.6394	5.6394	5.6394	5.6394	5.6394	5.6394	5.6393	5.6383	5.6271
0.700E 03	5.9753	5.9753	5.9753	5.9753	5.9753	5.9753	5.9751	5.9737	5.9580
0.100E 04	6.3315	6.3315	6.3315	6.3315	6.3315	6.3315	6.3313	6.3293	6.3069
0.150E 04	6.7367	6.7367	6.7367	6.7367	6.7367	6.7366	6.7363	6.7333	6.6997
0.200E 04	7.0242	7.0242	7.0242	7.0242	7.0242	7.0241	7.0237	7.0197	6.9750
0.300E 04	7.4295	7.4295	7.4295	7.4295	7.4295	7.4294	7.4287	7.4228	7.3561
0.500E 04	7.9402	7.9402	7.9402	7.9402	7.9402	7.9401	7.9389	7.9290	7.8192
0.700E 04	8.2766	8.2766	8.2766	8.2766	8.2766	8.2764	8.2748	8.2609	8.1092
0.100E 05	8.6332	8.6332	8.6332	8.6332	8.6332	8.6330	8.6307	8.6109	8.3983

FIGURE 4.4.—Example of output from program for computing drawdown due to constant discharge from a well in a leaky artesian aquifer.

Solution 5: Constant discharge from a well in a leaky aquifer with storage of water in the confining beds

Assumptions:

1. Well discharges at a constant rate, Q.
2. Well is of infinitesimal diameter and fully penetrates the aquifer.
3. Aquifer is overlain and underlain everywhere by confining beds having hydraulic conductivities K' and K'', thicknesses b' and b'', and storage coefficients S' and S'', respectively, which are constant in space and time.
4. Flow in the aquifer is two dimensional and radial in the horizontal plane and flow in confining beds is vertical. This assumption is approximated closely where the hydraulic conductivity of the aquifer is sufficiently greater than that of the confining beds.
5. Conditions at the far surfaces of the confining beds are (fig. 5.1):
   - Case 1. Constant-head plane sources above and below.
   - Case 2. Impermeable beds above and below.
   - Case 3. Constant-head plane source above and impermeable bed below.

Differential equations:

For the upper confining bed

$$\frac{\partial^2 s_1}{\partial z^2} = \frac{S'}{K'b'} \frac{\partial s_1}{\partial t} \tag{1}$$

For the aquifer

$$\frac{\partial^2 s}{\partial r^2} + \frac{1}{r}\frac{\partial s}{\partial r} + \frac{K'}{T}\frac{\partial}{\partial z} s_1(r,b',t) - \frac{K''}{T}\frac{\partial}{\partial z} s_2(r,b'+b,t) = \frac{S}{T}\frac{\partial s}{\partial t} \tag{2}$$

For the lower confining bed

$$\frac{\partial^2 s_2}{\partial z^2} = \frac{S''}{K''b''} \frac{\partial s_2}{\partial t} \tag{3}$$

Equations 1 and 3 are, respectively, the differential equations for nonsteady vertical flow in the upper and lower semipervious beds. Equation 2 is the differential equation for nonsteady two-dimensional radial flow in an aquifer with leakage at its upper and lower boundaries.

Boundary and initial conditions:

Case 1: For the upper confining bed

$$s_1(r,z,0)=0 \tag{4}$$
$$s_1(r,0,t)=0 \tag{5}$$
$$s_1(r,b',t)=s(r,t) \tag{6}$$

For the aquifer

$$s(r,0)=0 \tag{7}$$
$$s(\infty,t)=0 \tag{8}$$
$$\lim_{r\to 0} r\frac{\partial s(r,t)}{\partial r} = -\frac{Q}{2\pi T} \tag{9}$$

For the lower confining bed

$$s_2(r,z,0)=0 \tag{10}$$
$$s_2(r,b'+b+b'',t)=0 \tag{11}$$
$$s_2(r,b'+b,t)=s(r,t) \tag{12}$$

Case 2: Same as case 1, with conditions 5 and 11 being replaced, respectively, by

$$\frac{\partial s_1(r,0,t)}{\partial z} = 0 \tag{13}$$
$$\frac{\partial s_2(r,b'+b+b'')}{\partial z} = 0 \tag{14}$$

Case 3: Same as case 1, with condition 11 being replaced by condition 14.

Equations 4, 7, and 10 state that initially the drawdown is zero in the aquifer and within each confining bed. Equation 5 states that a plane of zero drawdown occurs at the top of the upper confining bed. Equations 6 and 12 state that, at the upper and lower boundaries of the aquifer, drawdown in the aquifer is equal to drawdown in the confining beds. Equation 8 states that drawdown is small at a large distance from the pumping well. Equation 9 states that, near the pumping well, the flow is equal to the discharge rate. Equation 11 states that a plane of zero drawdown is at the base of the lower confining bed. Equation 13 states that

there is no flow across the top of the upper confining bed. Equation 14 states that no flow occurs across the base of the lower confining bed.

Solutions (Hantush, 1960, p. 3716):

I. For small values of time (t less than both $b'S'/10K'$ and $b''S''/10K''$):

$$s = \frac{Q}{4\pi T} H(u,\beta) , \qquad (15)$$

where $$u = \frac{r^2 S}{4Tt}$$

and $$\beta = \frac{r}{4}\left(\sqrt{\frac{K'S'}{b'TS}} + \sqrt{\frac{K''S''}{b''TS}}\right)$$

$$H(u,\beta) = \int_u^\infty \frac{e^{-u}}{y}\, erfc\, \frac{\beta\sqrt{u}}{\sqrt{y(y-u)}}\, dy$$

$$erfc(x) = \frac{2}{\sqrt{\pi}}\int_x^\infty e^{-u^2}\, dy .$$

II. For large values of time:

A. Case 1, t greater than both $5b'S'/K'$ and $5b''S''/K''$

$$s = \frac{Q}{4\pi T} W(u\delta_1,\alpha) , \qquad (16)$$

where u is as defined previously

and $$\delta_1 = 1 + (S' + S'')/3S,$$

$$\alpha = r\sqrt{\frac{K'/b'}{T} + \frac{K''/b''}{T}}$$

$$W(u,x) = \int_u^\infty \frac{\exp(-y - x^2/4y)}{y}\, dy .$$

B. Case 2, t greater than both $10b'S'/K'$ and $10b''S''/K''$

$$s = \frac{Q}{4\pi T} W(u\delta_2) , \qquad (17)$$

where $$\delta_2 = 1 + (S' + S'')/S$$

$$W(u) = \int_u^\infty \frac{e^{-u}}{y}\, dy .$$

C. Case 3, t greater than both $5b'S'/K'$ and $10b''S''/K''$

$$s = \frac{Q}{4\pi T} W\left(u\delta_3,\ r\sqrt{\frac{K'/b'}{T}}\right) , \qquad (18)$$

where

$$\delta_3 = 1 + (S'' + S'/3)/S$$

and $W(u,x)$ is as defined in case 1.

Comments:

A cross section through the discharging well is shown in figure 5.1. The flow system is actually three-dimensional in such a geometric configuration. However, as stated by Hantush (1960, p. 3713), if the hydraulic conductivity in the aquifer is sufficiently greater than the hydraulic conductivity of the confining beds, flow will be approximately radial in the aquifer and approximately vertical in the confining beds. A complete solution to this flow problem has not been published. Neuman and Witherspoon (1971, p. 250, eq. II–161) developed a complete solution for case 1 but did not tabulate it. Hantush's solutions, which have been tabulated, are solutions that are applicable for small and large values of time but not for intermediate times.

The "early" data (data collected for small values of t) can be analyzed using equation 15. Figure 5.2 on plate 1 shows plots of $H(u,\beta)$ from Lohman (1972, pl. 4). Hantush (1961d) has an extensive tabulation of $H(u,\beta)$, a part of which is given in table 5.1. The corresponding data curves would consist of observed drawdown versus t/r^2. Superposing the data curves on the type curves and matching the two, with graph axes parallel, so that the data curves lie on or between members of the type-curve family and choosing a convenient match point ($H(u,\beta)$, $1/u$), T and S are computed by

$$T = \frac{Q}{4\pi s} H(u,\beta) ,$$

$$S = 4T\frac{t}{r^2}\Big/\frac{1}{u} .$$

If simplifying conditions are applicable, it is possible to compute the product $K'S'$ from the β value. If $K''S''=0$, $K'S'=16\beta^2 b'TS/r^2$, and if $K''S''=K'S'$,

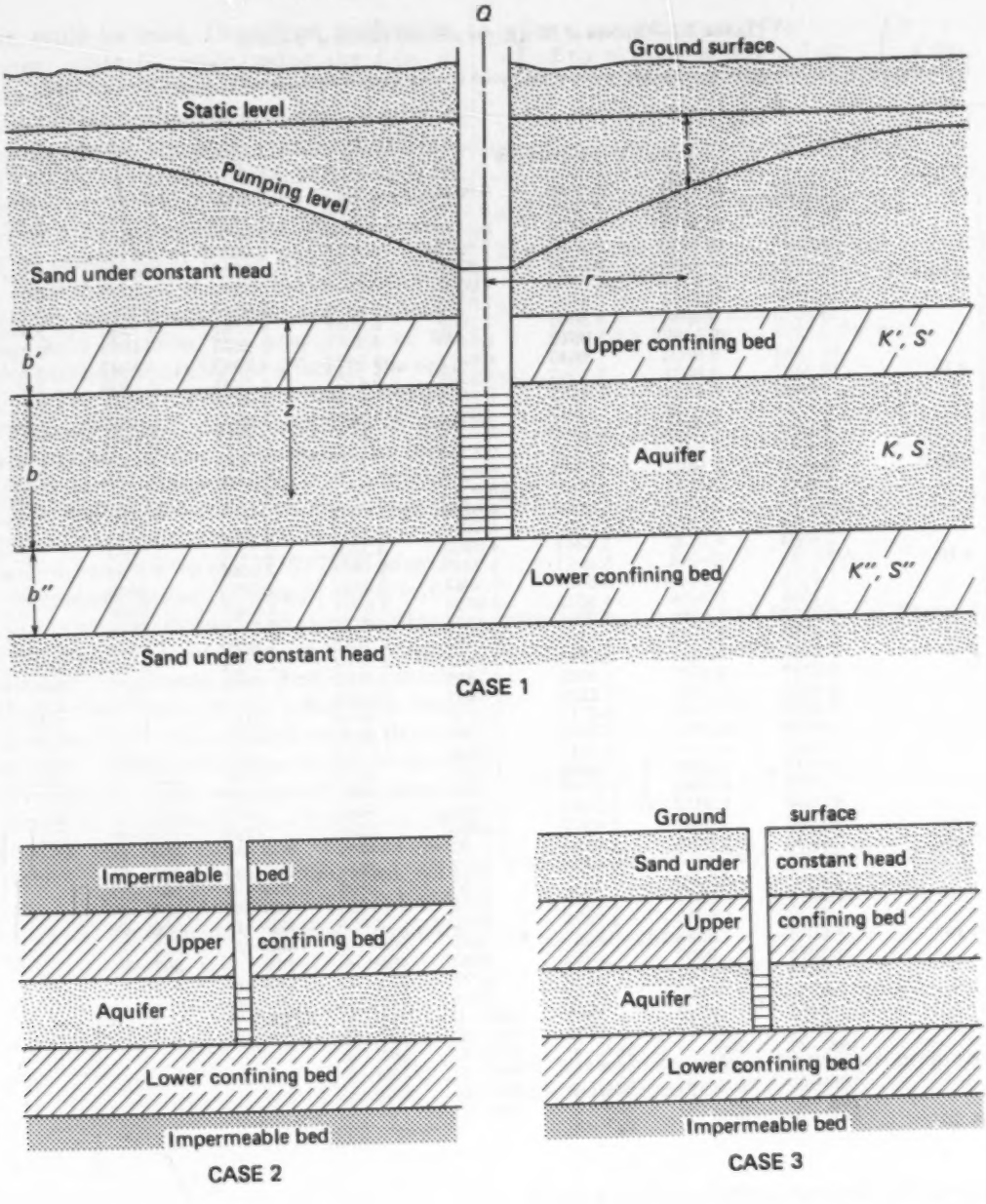

FIGURE 5.1.—Cross sections through discharging wells in leaky aquifers with storage of water in the confining beds, illustrating three different cases of boundary conditions.

$$K'S' = \frac{16\beta^2}{r^2} TS \frac{b'b''}{b'+b''+2\sqrt{b'b''}}.$$

The curves in figure 5.2 are very similar from $\beta=0$ to about $\beta=0.5$. Therefore, the β values in this range are indeterminate. There is also uncertainty in curve matching for all β values because of the fact that it is a family of curves whose shapes change gradually with β. This uncertainty will be increased if the data covers a small range of t values. The problem

TABLE 5.1.—*Values of* $W(u,\beta)$ *for selected values of* u *and* β

[From Hantush (1961d). Numbers in parentheses are powers of 10 by which the other numbers are multiplied; for example 963(−4) = 0.0963]

u	β							
	0.03	0.1	0.3	1	3	10	30	100
1×10^{-9}	12.3088	11.1051	10.0066	8.8030	7.7051	6.5033	5.4101	4.2221
2	11.9622	10.7585	9.6602	8.4566	7.3590	6.1579	5.0666	3.8839
3	11.7593	10.5558	9.4575	8.2540	7.1565	5.9561	4.8661	3.6874
5	11.5038	10.3003	9.2021	7.9987	6.9016	5.7020	4.6142	3.4413
7	11.3354	10.1321	9.0339	7.8306	6.7337	5.5348	4.4487	3.2804
1×10^{-8}	11.1569	9.9538	8.8556	7.6525	6.5558	5.3578	4.2737	3.1110
2	10.8100	9.6071	8.5091	7.3063	6.2104	5.0145	3.9352	2.7858
3	10.6070	9.4044	8.3065	7.1039	6.0085	4.8141	3.7383	2.5985
5	10.3511	9.1489	8.0512	6.8490	5.7544	4.5623	3.4919	2.3662
7	10.1825	8.9806	7.8830	6.6811	5.5872	4.3969	3.3307	2.2159
1×10^{-7}	10.0037	8.8021	7.7048	6.5032	5.4101	4.2221	3.1609	2.0591
2	9.6560	8.4554	7.3585	6.1578	5.0666	3.8839	2.8348	1.7633
3	9.4524	8.2525	7.1560	5.9559	4.8661	3.6874	2.6469	1.5966
5	9.1955	7.9968	6.9009	5.7018	4.6141	3.4413	2.4137	1.3944
7	9.0261	7.8283	6.7329	5.5346	4.4486	3.2804	2.2627	1.2666
1×10^{-6}	8.8463	7.6497	6.5549	5.3575	4.2736	3.1110	2.1051	1.1361
2	8.4960	7.3024	6.2091	5.0141	3.9350	2.7857	1.8074	.8995
3	8.2904	7.0991	6.0069	4.8136	3.7382	2.5984	1.6395	.7725
5	8.0304	6.8427	5.7523	4.5617	3.4917	2.3661	1.4354	.6256
7	7.8584	6.6737	5.5847	4.3962	3.3304	2.2158	1.3061	.5375
1×10^{-5}	7.6754	6.4944	5.4071	4.2212	3.1606	2.0590	1.1741	.4519
2	7.3170	6.1453	5.0624	3.8827	2.8344	1.7632	.9339	.3091
3	7.1051	5.9406	4.8610	3.6858	2.6464	1.5965	.8046	.2402
5	6.8353	5.6821	4.6075	3.4394	2.4131	1.3943	.6546	.1685
7	6.6553	5.5113	4.4408	3.2781	2.2619	1.2664	.5643	.1300
1×10^{-4}	6.4623	5.3297	4.2643	3.1082	2.1042	1.1359	.4763	963(−4)
2	6.0787	4.9747	3.9220	2.7819	1.8062	.8992	.3287	494(−4)
3	5.8479	4.7655	3.7222	2.5937	1.6380	.7721	.2570	315(−4)
5	5.5488	4.4996	3.4711	2.3601	1.4335	.6252	.1818	166(−4)
7	5.3458	4.3228	3.3062	2.2087	1.3039	.5370	.1412	103(−4)
1×10^{-3}	5.1247	4.1337	3.1317	2.0506	1.1715	.4513	.1055	390(−5)
2	4.6753	3.7598	2.7938	1.7516	.9305	.3084	551(−4)	169(−5)
3	4.3993	3.5363	2.5969	1.5825	.8006	.2394	355(−4)	713(−6)
5	4.0369	3.2483	2.3499	1.3767	.6498	.1677	190(−4)	205(−6)
7	3.7893	3.0542	2.1877	1.2460	.5589	.1292	120(−4)	821(−7)
1×10^{-2}	3.5195	2.8443	2.0164	1.1122	.4702	955(−4)	695(−5)	274(−7)
2	2.9759	2.4227	1.6853	.8677	.3214	487(−4)	205(−5)	226(−8)
3	2.6487	2.1680	1.4932	.7353	.2491	308(−4)	888(−6)	
5	2.2312	1.8401	1.2535	.5812	.1733	160(−4)	261(−6)	
7	1.9558	1.6213	1.0979	.4880	.1325	982(−5)	106(−6)	
1×10^{-1}	1.6667	1.3893	.9358	.3970	966(−4)	552(−5)	365(−7)	
2	1.1278	.9497	.6352	.2452	468(−4)	149(−5)	307(−8)	
3	.8389	.7103	.4740	.1729	281(−4)	592(−6)		
5	.5207	.4436	.2956	.1006	130(−4)	151(−6)		
7	.3485	.2980	.1985	646(−4)	714(−5)	534(−7)		
1×1	.2050	.1758	.1172	365(−4)	337(−5)	151(−7)		
2	458(−4)	395(−4)	264(−4)	760(−5)	487(−6)			
3	122(−4)	106(−4)	707(−5)	196(−5)	102(−6)			
5	108(−5)	934(−6)	624(−6)	167(−6)	672(−8)			
7	109(−6)	941(−7)	629(−7)	165(−7)				
1×10	391(−8)	339(−8)	227(−8)					
2								
3								
5								
7								

can be avoided, if data from more than one observation well are available, by preparing a composite data plot of s versus t/r^2. This data plot would be matched by adding the constraint that the r values for the different data curves representing each well fall on proportional β curves.

The "late" data (for large values of t) can be analyzed using equations 16, 17, and 18; these equations are forms of summaries 1, $W(u)$, and 4, $L(u, v)$. However, for cases 1 and 3, the late data fall on the flat part of the $L(u,v)$ curves and a time-drawdown plot match would be indeterminate. Thus, only a distance-drawdown

match could be used. Drawdown predictions, however, could be made using the $L(u, v)$ curves.

Assumption 5, that no drawdown occurs in the source beds, has been examined by Neuman and Witherspoon (1969a, p. 810, 811) for the situation in which two aquifers are separated by a less permeable bed. This is equivalent to case 3 with $K''=0$ and $S''=0$. They concluded that (1) $H(u,\beta)$, in the asymptotic solution for early times, would not be affected appreciably because the properties of the source bed have a negligible effect on the solution for $Tt/r^2S \leq 1.6\beta^2/(r/B)^4$, which is equivalent to $t \leq S'b'/10K'$, where $B=\sqrt{Tb'/K'}$; and (2) if $T_s > 100T$, where T_s represents the transmissivity of the source bed, it is probably justified to neglect drawdown in the unpumped aquifer.

Table 5.2 is a listing of a FORTRAN program for computing values of $H(u,\beta)$ for $u \geq 10^{-60}$ using a procedure devised and programed by S. S. Papadopulos. Input data for this program consists of three cards. The first card contains the beginning value of $1/u$, coded in columns 1–10, in format E10.5, and the ending (largest) value of $1/u$, coded in columns 11–20, in format E10.5. The next two cards contain 12 values of β, coded in columns 1–10, 11–20, . . . , and 71–80 on the first card and columns 1–10, 11–20, . . . , 31–40 on the second card, all in format E10.5. The function is evaluated as follows (S. S. Papadopulos, written commun., 1975):

$$H(u,\beta) = \int_u^\infty (e^{-u}/y)\, erfc\, (\beta\sqrt{u}/\sqrt{y(y-u)})\, dy$$

$$= \int_u^\infty f\, dy\,,$$

where f represents the integrand. For $\beta=0$, $H(u,\beta)=W(u)$, where $W(u)$ is the well function of Theis. Because $erfc(x) \leq 1$ for $x \geq 0$, it follows that $H(u,\beta) \leq W(u)$, and for $u>10$, $W(u) \approx 0$ and therefore for $u>10$, $H(u,\beta) \approx 0$. The tables of $H(u,\beta)$ indicate that $H(u,\beta) \approx 0$ for $\beta>1$ and $\beta^2 u > 300$. For an arbitrarily small value of u, the integral can be considered as the sum of three integrals

$$\int_u^\infty f\, dy = \int_u^{u_1} f\, dy + \int_{u_1}^{u_2} f\, dy + \int_{u_2}^\infty f\, dy\,,$$

where $u_2 = (u/2)(1 + \sqrt{1+10^{20}\beta^2/u})$,

and $u_1 = (u/2)(1 + \sqrt{1+0.025\,\beta^2/u})$.

The significance of u_2 and u_1 is that

$$erfc\, (\beta\sqrt{u}/\sqrt{y(y-u)}) \approx 1 \text{ for } u > u_2$$

and

$$erfc\, (\beta\sqrt{u}/\sqrt{y(y-u)}) \approx 0 \text{ for } u < u_1.$$

Therefore,

$$\int_u^{u_1} f\, dy \approx 0\,,$$

and

$$\int_{u_2}^\infty f\, dy \approx W(u_2),$$

where $W(u_2)$ is the well function of Theis. The function can be evaluated as

$$H(u,\beta) \approx W(u) \text{ for } u > u_2$$

$$H(u,\beta) \approx \int_u^{u_2} f\, dy + W(u_2) \text{ for } u_1 < u < u_2$$

and $$H(u,\beta) \approx \int_{u_1}^{u_2} f\, dy + W(u_2) \text{ for } u < u_1\,.$$

If $u_2 > 10$, then

$$\int_{u_1}^{u_2} f\, dy = \int_{u_1}^{10} f\, dy,\ W(u_2) \approx 0\,.$$

An example of output from this program is shown in figure 5.3.

Solution 6: Constant discharge from a partially penetrating well in a leaky aquifer

Assumptions:

1. Well discharges at a constant rate, Q.
2. Well is of infinitesimal diameter and is screened in only part of the aquifer.
3. Aquifer has radial-vertical anisotropy.

W(U,BETA)

1/U	BETA 0.30E-01	0.10E 00	0.30E 00	0.10E 01	0.30E 01
0.100E 02	1.6667	1.3894	0.9358	0.3970	0.0965
0.150E 02	1.9953	1.6531	1.1203	0.5010	0.1376
0.200E 02	2.2308	1.8401	1.2536	0.5812	0.1733
0.300E 02	2.5626	2.1010	1.4435	0.7023	0.2320
0.500E 02	2.9759	2.4228	1.6853	0.8677	0.3214
0.700E 02	3.2428	2.6296	1.8457	0.9836	0.3897
0.100E 03	3.5196	2.8443	2.0164	1.1122	0.4702
0.150E 03	3.8256	3.0826	2.2112	1.2647	0.5717
0.200E 03	4.0369	3.2483	2.3499	1.3767	0.6498
0.300E 03	4.3259	3.4775	2.5459	1.5394	0.7683
0.500E 03	4.6754	3.7598	2.7938	1.7516	0.9305
0.700E 03	4.8969	3.9425	2.9576	1.8953	1.0447
0.100E 04	5.1247	4.1338	3.1317	2.0507	1.1715
0.150E 04	5.3756	4.3486	3.3301	2.2306	1.3225
0.200E 04	5.5488	4.4996	3.4712	2.3602	1.4335
0.300E 04	5.7971	4.7109	3.6704	2.5452	1.5951
0.500E 04	6.0787	4.9747	3.9220	2.7819	1.8062
0.700E 04	6.2565	5.1474	4.0880	2.9396	1.9494
0.100E 05	6.4623	5.3297	4.2643	3.1082	2.1042
0.150E 05	6.6816	5.5361	4.4650	3.3014	2.2837
0.200E 05	6.8353	5.6821	4.6076	3.4394	2.4131
0.300E 05	7.0498	5.8874	4.8087	3.6349	2.5979
0.500E 05	7.3170	6.1454	5.0624	3.8827	2.8344
0.700E 05	7.4915	6.3149	5.2297	4.0467	2.9921
0.100E 06	7.6754	6.4944	5.4072	4.2212	3.1606
0.150E 06	7.8834	6.6983	5.6090	4.4202	3.3538
0.200E 06	8.0304	6.8427	5.7523	4.5617	3.4917
0.300E 06	8.2369	7.0462	5.9544	4.7616	3.6872
0.500E 06	8.4960	7.3024	6.2091	5.0141	3.9351
0.700E 06	8.6662	7.4710	6.3770	5.1807	4.0991
0.100E 07	8.8463	7.6497	6.5549	5.3576	4.2736
0.150E 07	9.0507	7.8528	6.7573	5.5589	4.4726
0.200E 07	9.1955	7.9968	6.9010	5.7018	4.6141
0.300E 07	9.3995	8.1998	7.1034	5.9035	4.8141
0.500E 07	9.6560	8.4554	7.3586	6.1578	5.0665
0.700E 07	9.8249	8.6237	7.5267	6.3255	5.2332
0.100E 08	10.0038	8.8022	7.7049	6.5039	5.4101
0.150E 08	10.2070	9.0050	7.9075	6.7055	5.6114
0.200E 08	10.3512	9.1489	8.0512	6.8490	5.7544
0.300E 08	10.5543	9.3517	8.2539	7.0513	5.9561
0.500E 08	10.8101	9.6072	8.5092	7.3063	6.2104
0.700E 08	10.9785	9.7754	8.6773	7.4744	6.3781
0.100E 09	11.1570	9.9538	8.8556	7.6525	6.5554
0.150E 09	11.3599	10.1566	9.0583	7.8550	6.7581
0.200E 09	11.5039	10.3004	9.2021	7.9988	6.9016
0.300E 09	11.7067	10.5032	9.4048	8.2014	7.1040
0.500E 09	11.9622	10.7586	9.6602	8.4566	7.3590
0.700E 09	12.1305	10.9269	9.8284	8.6248	7.5270
0.100E 10	12.3089	11.1052	10.0067	8.8031	7.7052

FIGURE 5.3.—Example of output from program for computing drawdown due to constant discharge from a well in a leaky aquifer with storage of water in the confining beds.

4. Aquifer is overlain, or underlain, everywhere by a confining bed having uniform hydraulic conductivity (K') and thickness (b').
5. Confining bed is overlain, or underlain, by an infinite constant-head plane source.
6. Hydraulic gradient across confining bed changes instantaneously with a change in head in the aquifer (no release of water from storage in the confining bed).
7. Flow is vertical in the confining bed.
8. The leakage from the confining bed is assumed to be generated within the aquifer so that in the aquifer no vertical flow results from leakage alone.

Differential equation:

$$\partial^2 s/\partial r^2 + 1/r\ \partial s/\partial r + a^2\partial^2 s/\partial z^2 - sK'/Tb' = S/T\ \partial s/\partial t$$

$$a^2 = K_z/K_r$$

This is the differential equation describing nonsteady radial and vertical flow in a homogeneous aquifer with radial-vertical anisotropy and leakage proportional to drawdown.

Boundary and initial conditions:

$$s(r,z,0) = 0,\ r \geq 0,\ 0 \leq z \leq b \quad (1)$$

$$s(\infty,z,t) = 0,\ 0 \leq z \leq b,\ t \geq 0 \quad (2)$$

$$\partial s(r,0,t)/\partial z = 0,\ r \geq 0,\ t \geq 0 \quad (3)$$

$$\partial s(r,b,t)/\partial z = 0,\ r \geq 0,\ t \geq 0 \quad (4)$$

$$\lim_{r\to 0} r\frac{\partial s}{\partial r} = \begin{cases} 0, & \text{for } 0 < z < d \\ -Q/(2\pi K_r(l-d)), & \text{for } d < z < l \\ 0, & \text{for } l < z < b \end{cases} \quad (5)$$

Equation 1 states that, initially, drawdown is zero. Equation 2 states that drawdown is small at a large distance from the pumping well. Equations 3 and 4 state that there is no vertical flow at the upper and lower boundaries of the aquifer. This means that vertical head gradients in the aquifer are caused by the geometric placement of the pumping well screen and not by leakage. Equation 5 states that near the pumping well the discharge is distributed uniformly over the well screen and that no radial flow occurs above and below the screen.

Solution:

I. For the drawdown in a piezometer, a solution by Hantush (1964a, p. 350) is given by

$$s = Q/4\pi T\{W(u,\beta) + f(u,ar/b,\beta,d/b,l/b,z/b)\},$$

where

$$W(u,\beta) = \int_u^\infty \frac{e^{-y-\frac{\beta^2}{4y^2}}}{y}\,dy$$

$$u = \frac{r^2 S}{4Tt}$$

$$\beta = \sqrt{\frac{r^2 K'}{Tb'}}$$

$$a = \sqrt{K_z/K_r}$$

$$f(u,ar/b,\beta,d/b,l/b,z/b) = 2b/\pi(l-d)\sum_{n=1}^{\infty} 1/n(\sin n\pi l/b - \sin n\pi d/b)\cdot\cos(n\pi z/b)W\left(u,\sqrt{\beta^2+(n\pi ar/b)^2}\right).$$

II. For the drawdown in an observation well

$$s = Q/4\pi T\{W(u,\beta) + \bar{f}(u,ar/b,\beta,d/b,l/b,d'/b,l'/b)\},$$

where

$$\bar{f}(u,ar/b,\beta,d/b,l/b,d'/b,l'/b) = 2b^2/\pi^2(l-d)(l'-d')\cdot\sum_{n=1}^{\infty} 1/n^2(\sin n\pi l/b - \sin n\pi d/b)\cdot(\sin n\pi l'/b - \sin n\pi d'/b)W(u,\sqrt{\beta^2 + (n\pi ar/b)^2})$$

Comments:

The geometry is shown in figure 6.1. The differential equation and boundary conditions are based on the assumption that vertical flow in the aquifer is caused by partial penetration of the pumping well and not by leakage. Hantush (1967, p. 587) concluded that this assumption is correct if $b\sqrt{K'/Tb'} < 0.1$. The solutions are based on a uniform distribution of flow over the screen of the pumped well. Depending on friction losses within the well, a more realistic assumption might be constant drawdown over

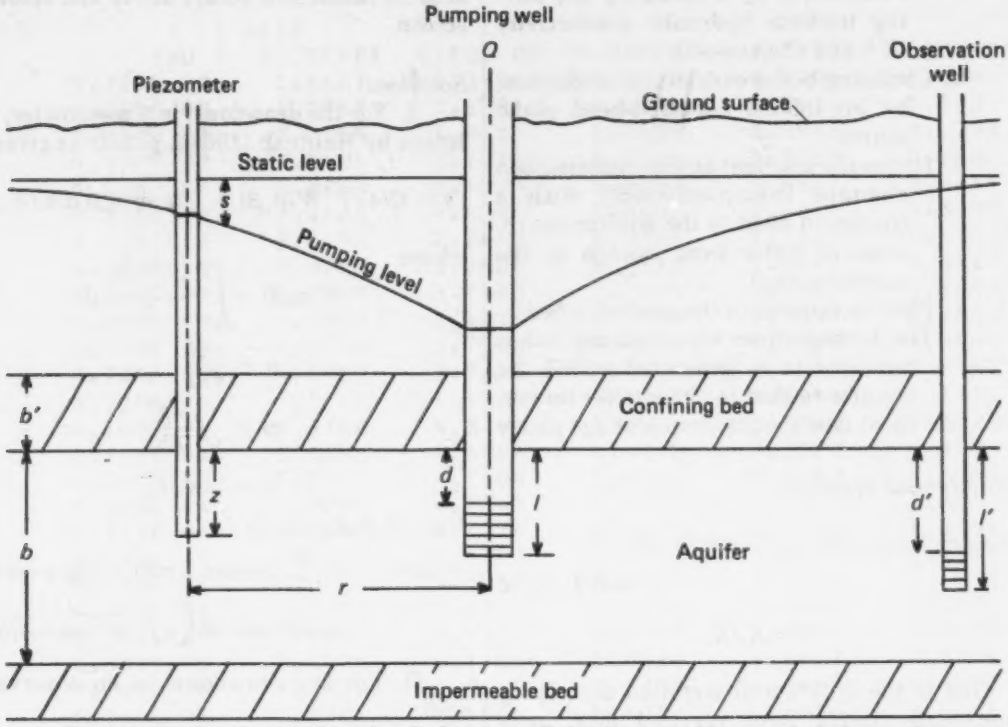

FIGURE 6.1.—Cross section through a discharging well that is screened in part of a leaky aquifer.

the screen of the pumped well; this assumption would imply nonuniform distribution of flow. Hantush (1964a, p. 351) postulates that the actual drawdown at the face of the pumping well will have a value between these two extremes. The solutions should be applied with caution at locations very near the pumped well. The effects of partial penetration are insignificant for $r>1.5\ b/a$ (Hantush, 1964a, p. 350), and the solution is the same for the solution 4.

Because of the large number of variables involved, presentation of a complete set of type curves is impractical. An example, consisting of curves for selected values of the parameters, is shown in figure 6.2 on plate 1. This figure is based on function values generated by a FORTRAN program.

The computer program formulated to compute drawdowns due to pumping a partially penetrating well in a leaky aquifer is listed in table 6.1. Input data to this program consists of cards coded in specific FORTRAN formats. Readers unfamiliar with FORTRAN format items should consult a FORTRAN language manual. The first card contains: aquifer thickness (b), coded in format F5.1 in columns 1–5; depth, below top of aquifer, to bottom of pumping well screen (l), coded in format F5.1 in columns 6–10; depth, below top of aquifer, to top of pumping well screen (d), coded in format F5.1 in columns 11–15; number of observation wells and piezometers, coded in format I5 in columns 16–20; smallest value of $1/u$ for which computation is desired, coded in format E10.4 in columns 21–30; largest value of $1/u$ for which computation is desired, coded in format E10.4 in columns 31–40. The next two cards contain 12 values of r/B, all coded in format E10.5, in columns 1–10, 11–20, 21–30, 31–40, 41–50, 51–60, 61–70, and 71–80 of the first card and columns 1–10, 11–20, 21–30, and 31–40 of the second card. Computation will terminate with the first zero (or blank) value coded. Next is a series of cards, one card per observation well or piezometer, containing: radial distance from the pumped well multiplied

by the square root of the ratio of vertical to horizontal conductivity ($r\sqrt{K_z/K_r}$), coded in format F5.1 in columns 1–5; depth, below top of aquifer, to bottom of observation well screen (code blank for piezometer), coded in format F5.1, in columns 6–10; depth, below top of aquifer, to top of observation well screen (total depth for a piezometer), coded in format F5.1, in columns 11–15. Output from this program is a table of function values. An example of the output is shown in figure 6.3.

Because most aquifers are anisotropic in the $r-z$ plane, it is generally impractical to use this solution to analyze for the parameters. However, it can be used to predict drawdown if the parameters are determined independently.

W(U,R/BR)*F(U,R/B,R/BR,L/B,D/B,Z/B), Z/B= 0.50, SQRT(KZ/KR)*R/B= 0.10, L/B= 0.70, D/B= 0.30

	R/BR							
1/U	0.10E-05	0.10E-04	0.10E-03	0.10E-02	0.10E-01	0.10E 00	0.10E 01	0.10E 02
0.100E 01	0.5478	0.5478	0.5478	0.5478	0.5478	0.5468	0.4631	0.0001
0.150E 01	0.9901	0.9901	0.9901	0.9901	0.9900	0.9878	0.7872	0.0001
0.200E 01	1.3804	1.3804	1.3804	1.3804	1.3803	1.3764	1.0398	0.0001
0.300E 01	2.0043	2.0043	2.0043	2.0043	2.0042	1.9964	1.3767	0.0001
0.500E 01	2.8381	2.8381	2.8381	2.8381	2.8379	2.8221	1.6931	0.0001
0.700E 01	3.3737	3.3737	3.3737	3.3737	3.3735	3.3499	1.8158	0.0001
0.100E 02	3.9049	3.9049	3.9049	3.9049	3.9046	3.8700	1.8826	0.0001
0.150E 02	4.4488	4.4488	4.4488	4.4488	4.4483	4.3975	1.9094	0.0001
0.200E 02	4.7951	4.7951	4.7951	4.7951	4.7944	4.7291	1.9143	0.0001
0.300E 02	5.2379	5.2379	5.2379	5.2379	5.2369	5.1455	1.9155	0.0001
0.500E 02	5.7539	5.7539	5.7539	5.7539	5.7525	5.6135	1.9155	0.0001
0.700E 02	6.0864	6.0864	6.0864	6.0864	6.0844	5.9001	1.9155	0.0001
0.100E 03	6.4390	6.4390	6.4390	6.4389	6.4363	6.1859	1.9155	0.0001
0.150E 03	6.8411	6.8411	6.8411	6.8411	6.8372	6.4816	1.9155	0.0001
0.200E 03	7.1271	7.1271	7.1271	7.1271	7.1220	6.6669	1.9155	0.0001
0.300E 03	7.5309	7.5309	7.5309	7.5309	7.5233	6.8854	1.9155	0.0001
0.500E 03	8.0404	8.0404	8.0404	8.0403	8.0278	7.0788	1.9155	0.0001
0.700E 03	8.3763	8.3763	8.3763	8.3762	8.3588	7.1556	1.9155	0.0001
0.100E 04	8.7326	8.7326	8.7326	8.7323	8.7076	7.2002	1.9155	0.0001
0.150E 04	9.1377	9.1377	9.1377	9.1373	9.1005	7.2199	1.9155	0.0001
0.200E 04	9.4252	9.4252	9.4252	9.4247	9.3758	7.2239	1.9155	0.0001
0.300E 04	9.8305	9.8305	9.8305	9.8298	9.7568	7.2250	1.9155	0.0001
0.500E 04	10.3412	10.3412	10.3412	10.3400	10.2199	7.2251	1.9155	0.0001
0.700E 04	10.6776	10.6776	10.6776	10.6759	10.5099	7.2251	1.9155	0.0001
0.100E 05	11.0343	11.0343	11.0343	11.0318	10.7990	7.2251	1.9155	0.0001

W(U,R/BR)*F(U,R/B,R/BR,L/B,D/B,L'/B,D'/B), L'/B= 0.51, D'/B= 0.49, SQRT(KZ/KR)*R/B= 0.10, L/B= 0.70, D/B= 0.30

	R/BR							
1/U	0.10E-05	0.10E-04	0.10E-03	0.10E-02	0.10E-01	0.10E 00	0.10E 01	0.10E 02
0.100E 01	0.5477	0.5477	0.5477	0.5477	0.5477	0.5468	0.4631	0.0001
0.150E 01	0.9899	0.9899	0.9899	0.9899	0.9899	0.9876	0.7871	0.0001
0.200E 01	1.3801	1.3801	1.3801	1.3801	1.3801	1.3761	1.0396	0.0001
0.300E 01	2.0038	2.0038	2.0038	2.0038	2.0037	1.9959	1.3764	0.0001
0.500E 01	2.8372	2.8372	2.8372	2.8372	2.8371	2.8213	1.6927	0.0001
0.700E 01	3.3727	3.3727	3.3727	3.3727	3.3725	3.3488	1.8153	0.0001
0.100E 02	3.9037	3.9037	3.9037	3.9037	3.9034	3.8688	1.8821	0.0001
0.150E 02	4.4475	4.4475	4.4475	4.4475	4.4470	4.3962	1.9089	0.0001
0.200E 02	4.7937	4.7937	4.7937	4.7937	4.7930	4.7277	1.9138	0.0001
0.300E 02	5.2365	5.2365	5.2365	5.2365	5.2356	5.1441	1.9150	0.0001
0.500E 02	5.7525	5.7525	5.7525	5.7525	5.7511	5.6122	1.9150	0.0001
0.700E 02	6.0850	6.0850	6.0850	6.0849	6.0830	5.8987	1.9150	0.0001
0.100E 03	6.4376	6.4376	6.4376	6.4375	6.4349	6.1845	1.9150	0.0001
0.150E 03	6.8397	6.8397	6.8397	6.8397	6.8358	6.4802	1.9150	0.0001
0.200E 03	7.1257	7.1257	7.1257	7.1257	7.1206	6.6655	1.9150	0.0001
0.300E 03	7.5295	7.5295	7.5295	7.5295	7.5219	6.8840	1.9150	0.0001
0.500E 03	8.0390	8.0390	8.0390	8.0389	8.0264	7.0775	1.9150	0.0001
0.700E 03	8.3749	8.3749	8.3749	8.3748	8.3574	7.1542	1.9150	0.0001
0.100E 04	8.7312	8.7312	8.7312	8.7309	8.7062	7.1988	1.9150	0.0001
0.150E 04	9.1363	9.1363	9.1363	9.1359	9.0991	7.2185	1.9150	0.0001
0.200E 04	9.4238	9.4238	9.4238	9.4233	9.3743	7.2225	1.9150	0.0001
0.300E 04	9.8291	9.8291	9.8291	9.8284	9.7554	7.2236	1.9150	0.0001
0.500E 04	10.3398	10.3398	10.3398	10.3386	10.2185	7.2237	1.9150	0.0001
0.700E 04	10.6762	10.6762	10.6762	10.6745	10.5085	7.2237	1.9150	0.0001
0.100E 05	11.0329	11.0329	11.0328	11.0304	10.7976	7.2237	1.9150	0.0001

FIGURE 6.3.—Example of output from program for partial penetration in a leaky artesian aquifer.

Solution 7: Constant drawdown in a well in a leaky aquifer

Assumptions:

1. Water level in well is changed instantaneously by s_w at $t=0$.
2. Well is of finite diameter and fully penetrates the aquifer.
3. Aquifer is overlain, or underlain, everywhere by a confining bed having uniform hydraulic conductivity (K') and thickness (b').
4. Confining bed is overlain, or underlain, by an infinite constant-head plane source.
5. Hydraulic gradient across confining bed changes instantaneously with a change in head in the aquifer (no release of water from storage in the confining bed).
6. Flow in the aquifer is two dimensional and radial in the horizontal plane and flow in the confining bed is vertical. This assumption is approximated closely where the hydraulic conductivity of the aquifer is sufficiently greater than that of the confining bed.

Differential equation:

$$\partial^2 s/\partial r^2 + (1/r)\partial s/\partial r - sK'/Tb' = (S/T)\partial s/\partial t$$

This differential equation describes nonsteady radial flow in a homogeneous isotropic confined aquifer with leakage proportional to drawdown.

Boundary and initial conditions:

$$s(r,0)=0,\ r\geq 0 \quad (1)$$
$$s(r_w,t)=s_w,\ t\geq 0 \quad (2)$$
$$s(\infty,t)=0,\ t\geq 0 \quad (3)$$

Equation 1 states that, initially, drawdown is zero. Equation 2 states that at the wall or screen of the discharging well, drawdown in the aquifer is equal to the constant drawdown in the well, which assumes that there is no entrance loss to the discharging well. Equation 3 states that the drawdown approaches zero as distance from the discharging well approaches infinity.

Solutions (Hantush, 1959):

I. For the discharge rate of the well,

$$Q = 2\pi T s_w G(\alpha, r_w/B),$$

where

$$G(\alpha,r_w/B) = (r_w/B)K_1(r_w/B)/K_0(r_w/B) + (4/\pi^2)\exp\left[-\alpha(r_w/B)^2\right] \cdot \int_0^\infty \left\{ u\exp(-\alpha u^2)/\left[J_0^2(u) + Y_0^2(u)\right]\right\} \cdot du/\left[u^2 + (r_w/B)^2\right],$$

and

$$\alpha = Tt/Sr_w^2,$$

$$B = \sqrt{Tb'/K'}.$$

K_0 and K_1 are zero-order and first-order, respectively, modified Bessel functions of the second kind. J_0 and Y_0 are the zero-order Bessel functions of the first and second kind, respectively.

II. For the drawdown in water level

$$s = s_w(K_0(r/B)/K_0(r_w/B) + (2/\pi)\exp(-\alpha r_w^2/B^2)\int_0^\infty \frac{\exp(-\alpha u^2)}{u^2 + (r_w/B)^2} \cdot \frac{J_0(ur/r_w)Y_0(u) - Y_0(ur/r_w)J_0(u)}{J_0^2(u) + Y_0^2(u)}\, u\, du \quad (4)$$

with α, B, K_0, J_0, and Y_0 as defined previously.

Comments:

A cross section through the discharging well is shown in figure 7.1. The boundary conditions most commonly apply to a flowing artesian well, as is shown in this illustration.

Figure 7.2 on plate 1 is a plot of dimensionless discharge ($G(\alpha,r_w/B)$) versus dimensionless time (α) from data of Hantush (1959, table 1) and Dudley (1970, table 2). Selected values of $G(\alpha,r_w/B)$ are given in table 7.1. The corresponding data curve should be a plot of observed discharge versus time. The data curve is matched to figure 7.2 and from match points $(\alpha,G(\alpha,r_w/B))$ and (t,Q), T and S are computed from the equations

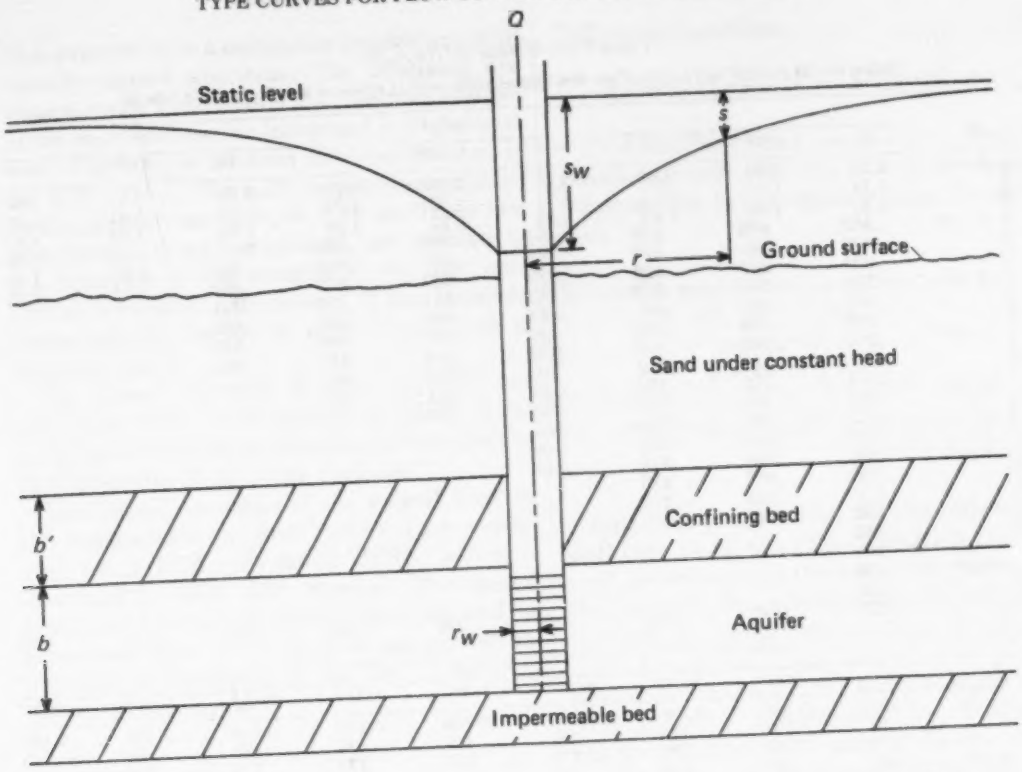

FIGURE 7.1.—Cross section through a well with constant drawdown in a leaky aquifer.

$$T = Q/(2\pi s_w G(\alpha, r_w/B))$$

and

$$S = Tt/(\alpha r_w^2).$$

Figure 7.3 on plate 1 contains plots of dimensionless drawdown (s/s_w) versus dimensionless time ($\alpha r_w^2/r^2$). The corresponding data plot would be observed drawdown versus observation time. Matching the data and type curves by superposition and choosing convenient match points $(s/s_w, \alpha r_w^2/r^2)$ and (s,t), the ratio of transmissivity to storage coefficient can be computed from the relation

$$T/S = (\alpha r_w^2/r^2)(r^2/t).$$

Figure 7.3 was plotted from function values generated by a FORTRAN program. This program is listed in table 7.2. The input data for this program consist of three cards coded in specific formats. Readers unfamiliar with FORTRAN format items should consult a FORTRAN language manual. The first card contains: the smallest value of alpha for which computation is desired, coded in format E10.5 in columns 1–10; the largest value of alpha for which computation is desired, coded in format E10.5 in columns 11–20. The output table will include a range in alpha spanning these two values up to a limiting range of nine log cycles. The second card contains 13 values of r_w/B. These coded values are the significant figures only and should be greater or equal to 1 and less than 10. The power of 10 by which each of these coded values is multiplied is calculated by the program. Zero (or blank) coding is permissible, but the first zero (or blank) value will terminate the list. The 13 values, all coded in format F5.0, are coded in columns 1–5, 6–10, 11–15, 16–20, 21–25, 26–30, 31–35, 36–40, 41–45, 46–50, 51–55, 56–60, and 61–65. The third card contains the radius of the control well and distances to the observation wells.

TABLE 7.1.—*Values of* $G(\alpha, r_w/B)$

[Values for $r_w/B \leq 1\times10^{-4}$ and $\alpha \geq 1\times10^{2}$ are from Hantush (1959, table 1), others are from Dudley (1970, table 2)]

α	r_w/B = 0	6×10^{-3}	1×10^{-2}	2×10^{-2}	6×10^{-2}	1×10^{-1}	2×10^{-1}	6×10^{-1}	1×10^{0}
1×10^{-1}	2.24	2.24	2.24	2.25	2.25	2.25	2.26	2.31	2.43
2	1.71	1.71	1.71	1.71	1.72	1.72	1.73	1.81	1.96
5	1.23	1.23	1.23	1.23	1.23	1.24	1.25	1.38	1.61
1×10^{0}	.983	.983	.983	.984	.986	.990	1.01	1.18	1.49
2	.800	.800	.800	.801	.804	.809	.834	1.07	1.44
5	.628	.628	.628	.629	.633	.642	.682	1.01	1.43
1×10^{1}	.534	.534	.534	.535	.541	.554	.611		
2	.461	.461	.461	.462	.472	.491	.569		
5	.389	.389	.389	.390	.407	.438	.548		
1×10^{2}	.346	.346	.346	.349	.374	.417	.545		
2	.311	.311	.312	.316	.353	.408			
5	.274	.275	.276	.284	.341	.406			
1×10^{3}	.251	.252	.255	.266	.339				
2	.232	.234	.239	.255					
5	.210	.215	.222	.249					
1×10^{4}	.196	.204	.216	.248					
2	.185	.197	.213						
5	.170	.192	.212						
1×10^{5}	.161	.191							
2	.152								
5	.143								
1×10^{6}	.136								
2	.130								
5	.123	.191	.212	.248	.339	.406	.545	1.01	1.43

α	r_w/B = 0	1×10^{-3}	2×10^{-3}	6×10^{-3}	1×10^{-2}	2×10^{-2}	6×10^{-2}	1×10^{-1}	2×10^{-1}
1×10^{4}	0.196	0.196	0.196	0.196	0.196	0.196	0.196	0.196	0.197
2	.185	.185	.185	.185	.185	.185	.185	.185	.185
5	.170	.170	.170	.170	.170	.170	.170	.170	.173
1×10^{5}	.161	.161	.161	.161	.161	.161	.162	.162	.167
2	.152	.152	.152	.152	.152	.152	.153	.155	.163
5	.143	.143	.143	.143	.143	.143	.144	.148	.161
1×10^{6}	.136	.136	.136	.136	.136	.137	.139	.144	.159
2	.130	.130	.130	.130	.130	.131	.135	.143	.159
5	.123	.123	.123	.123	.123	.124	.133	.142	.158
1×10^{7}	.118	.118	.118	.118	.118	.120			
2	.114	.114	.114	.114	.114	.116			
5	.108	.108	.108	.108	.110				
1×10^{8}	.104	.104	.104	.105	.108				
2	.100	.100	.101	.103	.107				
5	.0958	.0958	.0966	.102					
1×10^{9}	.0927	.0930	.0943						
2	.0899	.0906	.0927						
5	.0864	.0880	.0916						
1×10^{10}	.0838	.0867	.0914						
2	.0814	.0862							
5	.0785	.0860							
1×10^{11}	.0764	.0860	.0914	.102	.107	.116	.133	.142	.158
2									
5									

The control well radius (r_w) is coded first, in columns 1–8 in format F8.2. The distances (r) to the observation wells (maximum of nine) are coded next, in monotonic increasing order (smallest r first, largest r last), in columns 9–16, 17–24, 25–32, 33–40, 41–48, 49–56, 57–64, 65–72, and 73–80, all in format F8.2. If two or more observation wells have the same distance, this common distance should be coded only once, the function values will apply to all wells at the same distance from the control well. If the number of observation wells is less than nine, the remaining columns on the card should be left blank.

The integral in equation 4 is approximated by

$$\int_0^{\infty} f(u,\alpha,r_w/B)\,du \doteq \sum_{i=1}^{8000} f(-\Delta u/2 + i\Delta u,\alpha,r_w/B)\,\Delta u\ .$$

This expression is a composite quadrature with equally spaced abscissas. The abscissas are chosen at the midpoints of the intervals instead of the ends because the integrand is singular at $u=0$. The value of Δu used is related to α and is $\Delta u \leq 10^{-3}/\sqrt{\alpha}$. The r_w/B values then selected by the program satisfy $r_w/B \geq 10\,\Delta u$. These two constraints, though empirical, are related to the behavior of the integrand; the first constraint is related to the term $e^{-\alpha u^2}$ as u becomes large, and the second to $u/(u^2+(r_w/B)^2)$ as u becomes small.

The Bessel functions $K_0(r/B)$, $K_0(r_w/B)$ are evaluated by the IBM subroutine BESK. A description of this subroutine may be found in the IBM Scientific Subroutine Package.

The Bessel functions of the second kind in the integrand, $Y_0(u)$ and $Y_0(ur/r_w)$, are evaluated using IBM subroutine BESY, which is discussed in IBM SSP manual. The Bessel functions $J_0(u)$ and $J_0(ur/r_w)$ are evaluated for arguments less than four by a polynomial approximation consisting of the first 10 terms of the series expansion

$$J_0(x) = \sum_{n=0}^{7} (-1)^n (x^2/2)^n/(n!)^2.$$

For arguments greater than or equal to four, the asymptotic expansion is used

$$J_0(x) = P \cos(x - \pi/4) + Q \sin(x - \pi/4).$$

P and Q are calculated by the algorithm used in IBM subroutine BESY.

The output from this program consists of tables of function values, an example of which is shown in figure 7.4.

Solution 8: Constant discharge from a fully penetrating well of finite diameter in a nonleaky aquifer

Assumptions:

1. Well discharges at a constant rate, Q.
2. Well is of finite diameter and fully penetrates the aquifer.
3. Aquifer is not leaky.
4. Discharge from the well is derived from a depletion of storage in the aquifer and inside the well bore.

Differential equation:

$$\partial^2 s/\partial r^2 + (1/r)\partial s/\partial r = (S/T)\partial s/\partial t,\ r \geq r_w$$

This differential equation describes nonsteady radial flow in a homogeneous isotropic aquifer in the region outside the pumped well.

Boundary and initial conditions:

$$s(r_w, t) = s_w(t),\ t>0 \quad (1)$$

$$s(\infty, t) = 0,\ t>0 \quad (2)$$

$$s(r, 0) = 0,\ r \geq r_w \quad (3)$$

$$s_w(0) = 0 \quad (4)$$

$$(2\pi r_w T)\partial s(r_w, t)/\partial r - (\pi r_c^2)\partial s_w(t)/\partial t = -Q,\ t>0 \quad (5)$$

Equation 1 states that the drawdown at the well bore is equal to the drawdown inside the well, assuming that there is no entrance loss at the well face. Equation 2 states that drawdown is small at a large distance from the pumping well. Equations 3 and 4 state that, initially, drawdown in the aquifer and inside the well is zero. Equation 5 states that the discharge of the well is equal to the sum of the flow into the well and the rate of decrease in storage inside the well.

Solution (Papadopulos and Cooper, 1967; Papadopulos, 1967):

$$s = (Q/4\pi T)\,F(u,\alpha,\rho),$$

where

$$F(u,\alpha,\rho) = (8\alpha/\pi)\int_0^\infty \frac{\left[(1-\exp(-\beta^2\rho^2/4u)\right]\left[J_0(\beta\rho)A(\beta)-Y_0(\beta\rho)B(\beta)\right]}{\left[A(\beta)\right]^2 + \left[B(\beta)\right]^2}\,\frac{d\beta}{\beta^2},$$

and

$$B(\beta) = \beta J_0(\beta) - 2\alpha J_1(\beta),$$
$$A(\beta) = \beta Y_0(\beta) - 2\alpha Y_1(\beta),$$
$$u = r^2 S/4Tt,$$
$$\alpha = r_w^2 S/r_c^2,$$
and
$$\rho = r/r_w.$$

J_0 and Y_0, J_1 and Y_1, are zero-order and first-order Bessel functions of the first and second kind, respectively.

Z(ALPHA,R/RW,RW/B), R/RW= 100.

ALPHA \ RW/B	0.10E-03	0.15E-03	0.20E-03	0.30E-03	0.50E-03	0.70E-03	0.10E-02	0.15E-02	0.20E-02	0.30E-02	0.50E-02	0.70E-02	0.10E-01
0.100E 05	0.114	0.114	0.114	0.114	0.113	0.113	0.113	0.112	0.112	0.109	0.102	0.091	0.074
0.150E 05	0.142	0.142	0.142	0.141	0.141	0.141	0.141	0.140	0.138	0.134	0.122	0.107	0.082
0.200E 05	0.161	0.161	0.161	0.161	0.161	0.161	0.160	0.159	0.157	0.151	0.135	0.115	0.086
0.300E 05	0.188	0.188	0.188	0.188	0.188	0.188	0.187	0.184	0.181	0.173	0.150	0.123	0.088
0.500E 05	0.221	0.221	0.221	0.221	0.220	0.220	0.218	0.214	0.209	0.196	0.162	0.128	0.089
0.700E 05	0.242	0.242	0.242	0.241	0.241	0.240	0.237	0.232	0.225	0.208	0.167	0.130	0.089
0.100E 06	0.263	0.262	0.262	0.262	0.261	0.260	0.257	0.250	0.240	0.218	0.169	0.130	0.089
0.150E 06	0.285	0.285	0.285	0.284	0.283	0.281	0.277	0.267	0.254	0.225	0.170	0.130	0.089
0.200E 06	0.300	0.300	0.300	0.299	0.298	0.295	0.289	0.277	0.262	0.228	0.171	0.130	0.089
0.300E 06	0.321	0.321	0.320	0.319	0.317	0.313	0.305	0.289	0.269	0.231	0.171	0.130	0.089
0.500E 06	0.345	0.345	0.344	0.343	0.339	0.333	0.322	0.299	0.275	0.232	0.171	0.130	0.089
0.700E 06	0.360	0.360	0.359	0.357	0.352	0.344	0.330	0.303	0.276	0.232	0.171	0.130	0.089
0.100E 07	0.375	0.375	0.374	0.371	0.364	0.355	0.337	0.305	0.277	0.232	0.171	0.130	0.089
0.150E 07	0.391	0.391	0.389	0.386	0.376	0.364	0.342	0.306	0.277	0.232	0.171	0.130	0.089
0.200E 07	0.402	0.401	0.400	0.396	0.384	0.368	0.344	0.307	0.277	0.232	0.171	0.130	0.089
0.300E 07	0.417	0.416	0.414	0.408	0.392	0.373	0.345	0.307	0.277	0.232	0.171	0.130	0.089
0.500E 07	0.435	0.432	0.429	0.421	0.399	0.376	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.700E 07	0.445	0.442	0.438	0.427	0.401	0.376	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.100E 08	0.456	0.452	0.446	0.43[illegible]	0.403	0.377	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.150E 08	0.467	0.461	0.454	0.437	0.403	0.377	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.200E 08	0.474	0.467	0.458	0.439	0.404	0.377	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.300E 08	0.483	0.473	0.462	0.440	0.404	0.377	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.500E 08	0.492	0.479	0.465	0.440	0.404	0.377	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.700E 08	0.497	0.482	0.466	0.440	0.404	0.377	0.346	0.307	0.277	0.232	0.171	0.130	0.089
0.100E 09	0.501	0.483	0.467	0.440	0.404	0.377	0.346	0.307	0.277	0.232	0.171	0.130	0.089

FIGURE 7.4.—Example of output from program for constant drawdown in a well in a leaky artesian aquifer.

The drawdown inside the pumped well is obtained at $r = r_w$ and can be expressed as (Papadopulos and Cooper, 1967, p. 242):

$$s_w = (Q/4\pi T)\, F(u_w,\alpha),$$

where $$F(u_w,\alpha) = F(u,\alpha,1),$$

and $$u_w = r_w^2 S/4tT.$$

Comments: A cross section through the discharging well is shown in figure 8.1. The geometry, except for the region of the well bore, is the same as for solution 1 (Theis solution). It is apparent from figure 8.2 and 8.3 (on plate 1) that $F(u,\alpha,\rho)$ approaches $W(u)$, the Theis solution, as time becomes large. Papadopulos (1967, p. 161) stated that for $t > 2.5\times10^3 r_c/T$, or $\alpha\rho^2/u > 10^4$, the function $F(u,\alpha,\rho)$ can be closely approximated by $F(u,\alpha,\rho) = W(u)$. Papadopulos and Cooper (1967, p. 242) stated that for $t > 2.5\times10^2\, r_c^2/T$, or $\alpha/u_w > 10^3$, the function $F(u_w,\alpha)$ can be closely approximated by $F(u_w,\alpha) = W(u_w)$. An examination of the type curves and function values indicates that $F(u_w,\alpha) \approx W(u_w)$ (less than 5-percent error) for $\alpha/u_w > 10^2$, and hence t should only be greater than $25\, r_c^2/T$ for drawdown in the pumped well.

Figures 8.2 and 8.3 were prepared from function values given in Papadopulos and Cooper (1967) and Papadopulos (1967), which are reproduced in table 8.1. For drawdown observations in the pumped well, the method of analysis is to plot drawdown versus time and

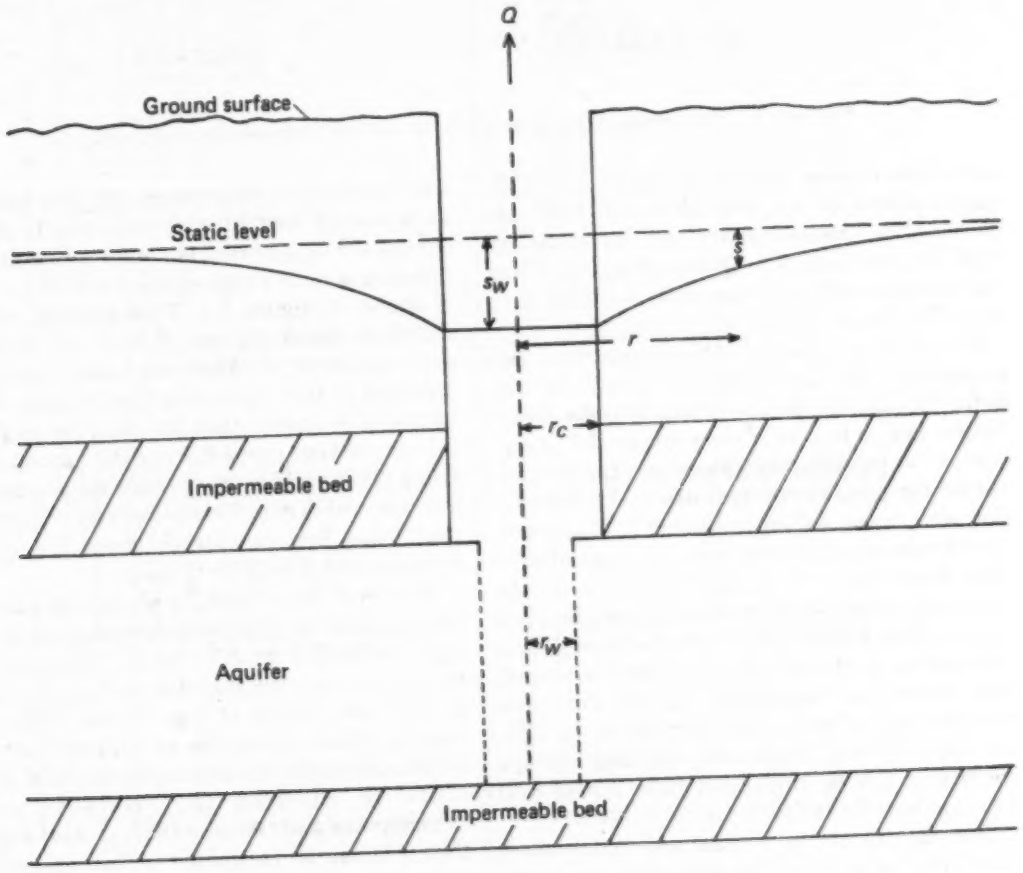

FIGURE 8.1.—Cross section through a discharging well of finite diameter.

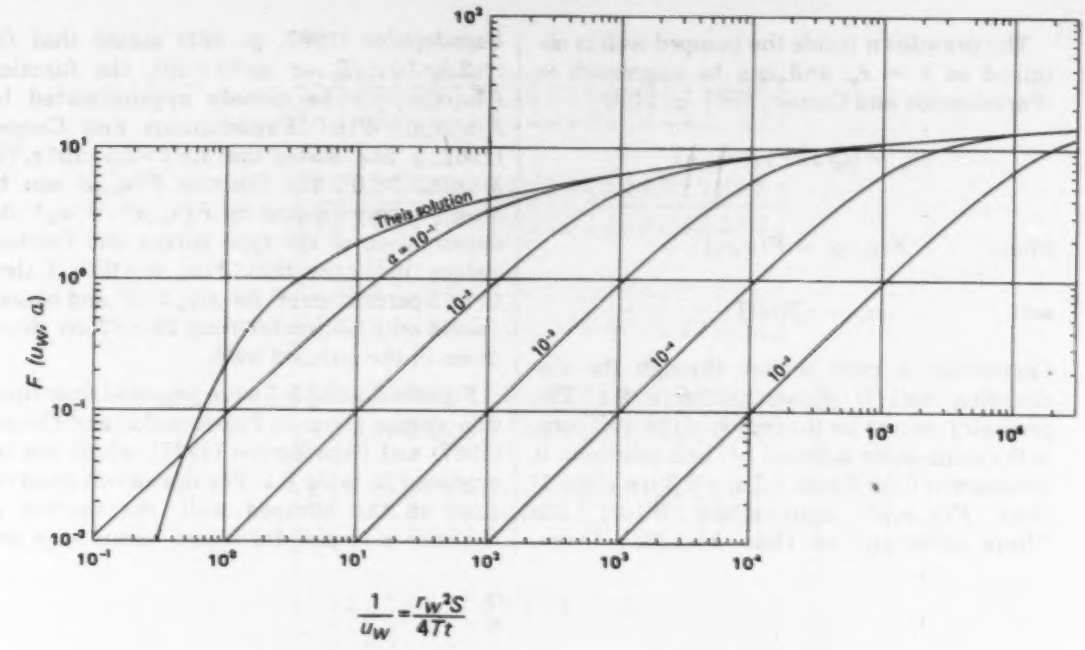

FIGURE 8.2.—Five selected type curves of $F(u_w,\alpha)$, and the Theis solution, versus $1/u_w$.

then superimpose the plot on figure 8.2. After match points of (s,t) and $(F(u_w,\alpha),\ 1/u_w)$ are chosen, the transmissivity can be computed from the relation $T=(Q/4\pi s)\ F(u_w,\alpha)$. Then, the storage coefficient can be determined from $S=(4\,Tt/r_w^2)/(1/u_w)$.

For observations not in the pumped well, two procedures are available for analyzing the data. To analyze the data from a single observation well, a family of type curves of $F(u,\alpha,\rho)$ versus $1/u$ for different values of α can be plotted for the ρ value appropriate for the observation well, using values in table 8.1. This procedure produces a family of type curves similar to that shown for $\rho=1$ in figure 8.2. If ρ for the observation well is between ρ values in table 8.1, function values can be interpolated. Using this approach, the data for the observation well are plotted as drawdown versus time and matched to the best-fitting member of the plotted type curves. Transmissivity and storage coefficient can be calculated from $T=(Q/4\pi s)\ F(u,\alpha,\rho)$ and $S=(4Tt/r^2)/(1/u)$.

Drawdowns at more than one observation point may be combined by preparing a composite plot of the drawdowns at each observation well versus t/r^2. This composite plot would be analyzed by matching it to a family of type curves of $F(u,\alpha,\rho)$ versus $1/u$ for constant α. An example of such a type-curve family for $\alpha=10^{-4}$ is shown in figure 8.3. This method requires multiple sheets of type curves, one sheet for each value of α. When the data curves are matched to the type-curve family, care should be taken to insure that the data for each well fall on the type curve having the appropriate ρ value. This will be possible for all the data for only one value of α. Transmissivity and storage coefficient are calculated from $T=(Q/4\pi s)\ F(u,\alpha,\rho)$ and $S=4T(t/r^2)/(1/u)$.

In both of these methods of plotting and comparing data, an alternate computation of storage coefficient is $S=r_c^2\alpha/r_w^2$. However, as pointed out by Papadopulos and Cooper (1967, p. 244), the shapes of type curves differ only slightly when α changes by an order of magnitude, therefore the determination of S is sensitive to choosing the "correct" curve. Papadopulos and Cooper (1967, p. 244) suggest that if S can be estimated within an order of magnitude, the value of α to be used for matching the data can be decided.

TABLE 8.1.—*Values of the function* $W(u,\alpha,\rho)$

[Values for $\rho = 1$ from Papadopulos and Cooper, 1967. Other values from Papadopulos, 1967]

u	ρ = 1	2	5	10	20	50	100	200
				For $\alpha = 10^{-1}$				
2×10^{0}	4.88×10^{-2}	1.96×10^{-2}	1.75×10^{-2}	2.41×10^{-2}	3.48×10^{-2}	4.24×10^{-2}	4.48×10^{-2}	4.50×10^{-2}
1	9.19	7.01	9.55	1.41×10^{-1}	1.85×10^{-1}	2.09×10^{-1}	2.14×10^{-1}	2.15×10^{-1}
5×10^{-1}	1.77×10^{-1}	1.95×10^{-1}	3.21×10^{-1}	4.44	5.20	5.49	5.55	5.59
2	4.06	5.78	9.42	1.13×10^{0}	1.19×10^{0}	1.22×10^{0}		
1	7.34	1.11×10^{0}	1.60×10^{0}	1.76	1.80			
5×10^{-2}	1.26×10^{0}	1.84	2.33	2.43	2.46			
2	2.30	2.97	3.28	3.34	3.35			
1	3.28	3.81	4.00	4.03				
5×10^{-3}	4.26	4.60	4.70	4.72				
2	5.42	5.58	5.63	5.64				
1	6.21	6.30	6.33					
5×10^{-4}	6.96	7.01						
2	7.87	7.93						
1	8.57	8.63						
5×10^{-5}	9.32							
2	10.24							
				For $\alpha = 10^{-2}$				
2×10^{0}	4.99×10^{-3}	2.13×10^{-3}	2.11×10^{-3}	3.52×10^{-3}	7.47×10^{-3}	2.03×10^{-2}	3.44×10^{-2}	4.35×10^{-2}
1	9.91	7.99	1.32×10^{-2}	2.69×10^{-2}	6.12×10^{-2}	1.42×10^{-1}	1.91×10^{-1}	2.11×10^{-1}
5×10^{-1}	1.97×10^{-2}	2.40×10^{-2}	5.40	1.21×10^{-1}	2.63×10^{-1}	4.65	5.31	5.51
2	4.89	8.34	2.33×10^{-1}	5.12	9.15	1.16×10^{0}	1.20×10^{0}	1.22×10^{0}
1	9.67	1.93×10^{-1}	5.67	1.12×10^{0}	1.58×10^{0}	1.78	1.81	
5×10^{-2}	1.90×10^{-1}	4.16	1.18×10^{0}	1.95	2.32	2.44	2.46	
2	4.53	1.03×10^{0}	2.42	3.11	3.29	3.34	3.35	
1	8.52	1.87	3.48	3.90	4.00	4.03		
5×10^{-3}	1.54×10^{0}	3.05	4.43	4.65	4.71	4.72		
2	3.04	4.78	5.52	5.61	5.63	5.64		
1	4.55	5.90	6.27	6.31	6.33			
5×10^{-4}	6.03	6.81	6.99	7.01				
2	7.56	7.85	7.92	7.94				
1	8.44	8.59	8.63					
5×10^{-5}	9.23	9.30						
2	10.20	10.23						
1	10.87	10.93						
5×10^{-6}	11.62	11.63						
2	12.54							
1	13.24							

TABLE 8.1.—*Values of the function* $F(u,\alpha,\rho)$—Continued

u	ρ = 1	2	5	10	20	50	100	200
				For $\alpha = 10^{-3}$				
2×10^{0}	5.00×10^{-4}	2.15×10^{-4}	2.15×10^{-4}	3.70×10^{-4}	8.35×10^{-3}	3.05×10^{-3}	8.38×10^{-3}	1.50×10^{-2}
1	9.99	8.11	1.37×10^{-3}	2.95×10^{-3}	7.58×10^{-3}	2.81×10^{-2}	7.56×10^{-3}	1.47×10^{-1}
5×10^{-1}	2.00×10^{-3}	2.45×10^{-3}	5.77	1.42×10^{-2}	3.90×10^{-2}	$1.54 \times ^{-1}$	3.23×10^{-1}	4.78
2	4.99	8.71	2.67×10^{-2}	7.24	2.03×10^{-1}	6.59	1.02×10^{0}	1.17×10^{0}
1	9.97	2.07×10^{-2}	7.16	2.01×10^{-1}	5.41	1.38×10^{0}	1.70	1.79
5×10^{-2}	1.99×10^{-2}	4.66	1.74×10^{-1}	4.87	1.19×10^{0}	2.27	2.40	2.45
2	4.95	1.29×10^{-1}	5.05	1.31×10^{0}	2.52	3.22	3.32	3.35
1	9.83	2.70	1.04×10^{0}	2.38	3.59	3.96	4.02	
5×10^{-3}	1.95×10^{-1}	5.47	1.96	3.68	4.50	4.69	4.72	
2	4.73	1.31×10^{0}	3.81	5.23	5.55	5.63	5.64	
1	9.07	2.39	5.34	6.13	6.28	6.32		
5×10^{-4}	1.69×10^{0}	3.98	6.57	6.92	7.00	7.02		
2	3.52	6.44	7.77	7.90	7.93			
1	5.53	7.95	8.55	8.61	8.63			
5×10^{-5}	7.63	9.02	9.28	9.31				
2	9.68	10.12	10.22	10.24				
1	10.68	10.88	10.93					
5×10^{-6}	11.50	11.59	11.62					
2	12.49	12.53	12.54					
1	13.21	13.23	13.24					
5×10^{-7}	13.92	13.93						
2	14.84							
1	15.54							
				For $\alpha = 10^{-4}$				
2×10^{0}	5.00×10^{-5}	2.17×10^{-5}	2.18×10^{-5}	3.73×10^{-5}	8.46×10^{-5}	3.16×10^{-4}	9.56×10^{-4}	3.83×10^{-3}
1	1.00×10^{-4}	8.15	1.38×10^{-4}	2.98×10^{-4}	7.77×10^{-4}	3.23×10^{-3}	1.01×10^{-2}	3.42×10^{-2}
5×10^{-1}	2.00	2.47×10^{-4}	5.81	1.45×10^{-3}	4.10×10^{-3}	1.80×10^{-2}	5.62	1.75×10^{-1}
2	5.00	8.76	2.71×10^{-3}	7.54	2.27×10^{-2}	1.03×10^{-1}	3.04×10^{-1}	7.10
1	1.00×10^{-3}	2.09×10^{-3}	7.34	2.16×10^{-2}	6.69	2.97	7.92	1.43×10^{0}
5×10^{-2}	2.00	4.72	1.82×10^{-2}	5.55	1.74×10^{-1}	7.30	1.62×10^{0}	2.24
2	5.00	1.32×10^{-2}	5.56	1.74×10^{-1}	5.36	1.87×10^{0}	2.95	3.28
1	9.98	2.81	1.23×10^{-1}	3.86	1.14×10^{0}	3.08	3.84	4.02
5×10^{-3}	1.99×10^{-2}	5.88	2.64	8.13	2.17	4.25	4.63	4.71
2	4.97	1.53×10^{-1}	6.89	1.97×10^{0}	4.14	5.47	5.60	5.63
1	9.90	3.10	1.36×10^{0}	3.44	5.61	6.24	6.31	6.33
5×10^{-4}	1.97×10^{-1}	6.18	2.53	5.26	6.71	6.98	7.01	
2	4.81	1.48×10^{0}	4.95	7.33	7.82	7.92	7.94	
1	9.34	2.72	7.03	8.37	8.57	8.62		
5×10^{-5}	1.77×10^{0}	4.65	8.65	9.20	9.29	9.32		
2	3.83	7.87	10.02	10.19	10.23	10.24		
1	6.25	9.92	10.83	10.91	10.93			
5×10^{-6}	8.99	11.23	11.57	11.62	11.63			

2	11.74	12.40	12.52	12.54
1	12.91	13.17	13.23	13.24
5×10^{-7}	13.78	13.90	13.93	
2	14.79	14.83		
1	15.51	15.53		
5×10^{-8}	16.22	16.23		
2	17.14			
1	17.84			

For $\alpha = 10^{-5}$

2×10^{0}	5.00×10^{-6}	2.27×10^{-6}	2.48×10^{-6}	4.19×10^{-6}	9.00×10^{-6}	3.21×10^{-5}	9.77×10^{-5}	3.15×10^{-4}
1	1.00×10^{-5}	8.36	1.44×10^{-5}	3.07×10^{-5}	7.89×10^{-5}	3.27×10^{-4}	1.04×10^{-3}	3.44×10^{-3}
5×10^{-1}	2.00	2.51×10^{-5}	5.94	1.47×10^{-4}	4.14×10^{-4}	1.84×10^{-3}	6.02	2.00×10^{-2}
2	5.00	8.87	2.74×10^{-4}	7.61	2.31×10^{-3}	1.08×10^{-2}	3.61×10^{-2}	1.19×10^{-1}
1	1.00×10^{-4}	2.11×10^{-4}	7.42	2.18×10^{-3}	6.85	3.30	1.10×10^{-1}	3.50
5×10^{-2}	2.00	4.77	1.84×10^{-3}	5.65	1.82×10^{-2}	8.90	2.92	8.57
2	5.00	1.34×10^{-3}	5.64	1.80×10^{-2}	5.92	2.89×10^{-1}	8.91	2.12×10^{0}
1	1.00×10^{-3}	2.84	1.26×10^{-2}	4.09	1.36×10^{-1}	6.49	1.80×10^{0}	3.34
5×10^{-3}	2.00	5.96	2.74	9.03	3.01	1.35×10^{0}	3.14	4.40
2	5.00	1.56×10^{-2}	7.43	2.47×10^{-1}	8.06	3.03	5.01	5.52
1	9.99	3.20	1.55×10^{-1}	5.15	1.60×10^{0}	4.75	6.06	6.27
5×10^{-4}	2.00×10^{-2}	6.54	3.20	1.04×10^{0}	2.96	6.31	6.90	6.99
2	4.98	1.66×10^{-1}	8.08	2.45	5.58	7.71	7.89	7.93
1	9.93	3.34	1.58×10^{0}	4.28	7.54	8.52	8.61	8.63
5×10^{-5}	1.98×10^{-1}	6.62	2.93	6.63	8.90	9.21	9.31	
2	4.86	1.59×10^{0}	5.86	9.36	10.10	10.22	10.24	
1	9.49	2.95	8.53	10.60	10.86	10.92		
5×10^{-6}	1.82×10^{0}	5.15	10.67	11.48	11.59	11.62		
2	4.03	9.08	12.28	12.49	12.53	12.54		
1	6.78	11.76	13.12	13.21	13.23	13.24		
5×10^{-7}	10.13	13.41	13.88	13.92	13.93			
2	13.71	14.68	14.83	14.85				
1	15.13	15.46	15.54					
5×10^{-8}	16.05	16.20						
2	17.08	17.14						
1	17.81	17.84						
5×10^{-9}	18.51							
2	19.40							
1	20.15							

The early parts (short time) of the curves in figure 8.2 are straight lines. According to Papadopulos and Cooper (1967, p. 244), these represent conditions under which all the water pumped is derived from storage within the well. The straight lines approached by the curves satisfy the equations

$$F(u_w,\alpha) = \alpha/u_w$$

and

$$s_w = Qt/\pi r_c^2 = \frac{\text{volume of water discharged}}{\text{area of well}}$$

Therefore, as pointed out by Papadopulos and Cooper (1067, p. 244), data that fall on this straight part of the type curves do not indicate information about the aquifer characteristics.

Table 8.2 is a listing of two FORTRAN programs by S. S. Papadopulos that evaluate $F(u_w,\alpha)$ and $F(u,\alpha,\rho)$. The input data to both programs consists of cards coded in specified format (readers unfamiliar with FORTRAN language format should refer to a FORTRAN language manual). Input to the programs is one or more groups of data, each group of data consisting of two cards. The first card contains one value of alpha in columns 1–10, coded in format E10.5. The program to evaluate $F(u,\alpha,\rho)$ also requires a value of rho on this card in columns 11–20. This value of rho, which must be greater than one, is also coded in format E10.5. The second card contains 16 values of u coded in columns 1–5, 6–10, . . . , 75–80 in format 16F5.0. The $F(u_w,\alpha)$ or $F(u,\alpha,\rho)$ values will be printed in the order that the u values are coded. If less than 16 values of u are desired, the remaining columns on the card may be left blank. Outputs from these two programs are shown in figures 8.4 and 8.5.

F(UW,ALPHA) FOR ALPHA= 1.00000E-04

UW	INTEGRAL	INTEGRAL ERROR	F(UW,ALPHA)	X(PEAK)	Y(PEAK)
2.00000E 00	1.54210E 03	-6.98844E-02	4.99991E-05	5.96561E-03	5.55886E 05
1.00000E 00	3.08412E 03	-1.39817E-01	9.99956E-05	5.96561E-03	1.11177E 06
5.00000E-01	6.16789E 03	-2.74775E-01	1.99980E-04	5.96561E-03	2.22353E 06
2.00000E-01	1.54184E 04	-6.97533E-01	4.99907E-04	5.96561E-03	5.55875E 06
1.00000E-01	3.08331E 04	-1.39715E 00	9.99695E-04	5.96560E-03	1.11173E 07
5.00000E-02	6.16529E 04	-2.71364E 00	1.99896E-03	5.96559E-03	2.22335E 07
2.00000E-02	1.54061E 05	-6.97112E 00	4.99507E-03	5.96559E-03	5.55764E 07
1.00000E-02	3.07919E 05	-1.39383E 01	9.98359E-03	5.96554E-03	1.11128E 08
5.00000E-03	6.15138E 05	-2.78767E 01	1.99445E-02	5.96549E-03	2.22157E 08
2.00000E-03	1.53334E 06	-6.82757E 01	4.97152E-02	5.96527E-03	5.54652E 08
1.00000E-03	3.05367E 06	-1.38658E 02	9.90083E-02	5.96493E-03	1.10684E 09
5.00000E-04	6.06085E 06	-2.76458E 02	1.96509E-01	5.96425E-03	2.20389E 09
2.00000E-04	1.48475E 07	-6.79220E 02	4.81397E-01	5.96223E-03	5.43712E 09
1.00000E-04	2.88072E 07	-1.30780E 03	9.34008E-01	5.95886E-03	1.06380E 10
5.00000E-05	5.45352E 07	-2.50960E 03	1.76818E 00	5.95237E-03	2.03734E 10
2.00000E-05	1.18065E 08	-5.40026E 03	3.82800E 00	5.93415E-03	4.49196E 10

FIGURE 8.4.—Example of output from program for drawdown inside a well of finite diameter due to constant discharge.

F(U,ALPHA,RHO) FOR ALPHA= 1.00000E-05, RHO= 2.00000E 00

U	INTEGRAL	INTEGRAL ERROR	F(U,ALPHA,RHO)
9.99999900E-04	6.29273600E 02	5.45096700E-01	3.20486300E-02
5.00000000E-04	1.28359500E 03	1.11649700E 00	6.53728800E-02
1.99999900E-04	3.26376700E 03	2.47402200E 00	1.66222200E-01
1.00000000E-04	6.55423000E 03	3.31468400E 00	3.33803700E-01
5.00000000E-05	1.30015800E 04	3.53750700E 00	6.62164900E-01
2.00000000E-05	3.11692500E 04	3.54940500E 00	1.58743500E 00
9.99999900E-06	5.79505700E 04	3.54602200E 00	2.95139600E 00
4.99999900E-06	1.01023500E 05	3.53222000E 00	5.14508300E 00
1.99999900E-06	1.78237100E 05	3.62180400E 00	9.07753300E 00
1.00000000E-06	2.30897600E 05	3.66347000E 00	1.17595100E 01
4.99999900E-07	2.63222100E 05	3.68847000E 00	1.34057800E 01
1.99999900E-07	2.88201800E 05	3.52180300E 00	1.46779900E 01

FIGURE 8.5.—Example of output from program for drawdown outside a well of finite diameter due to constant discharge.

Solution 9: Slug test for a finite-diameter well in a nonleaky aquifer

Assumptions:

1. A volume of water, V, is injected into, or is discharged from, the well instantaneously at $t=0$.
2. Well is of finite diameter and fully penetrates the aquifer.
3. Aquifer is not leaky, and flow is in radial direction only.

Differential equation:

$$\partial^2 h/\partial r^2 + (1/r)\,\partial h/\partial r = (S/T)\,\partial h/\partial t,\ r>r_w$$

This differential equation describes nonsteady radial flow in a homogeneous isotropic aquifer beyond the radius of the injected well.

Boundary and initial conditions:

$$h(r_w,t)=H(t),\ t>0 \quad (1)$$

$$h(\infty,t)=0,\ t>0 \quad (2)$$

$$2\pi r_w T\,\frac{\partial h(r_w,t)}{\partial r} = \pi r_c^2\,\frac{\partial H(t)}{\partial t},\ t>0 \quad (3)$$

$$h(r,0)=0,\ r>r_w \quad (4)$$

$$H(0) = H_0 = V/\pi r_c^2 \quad (5)$$

Equation 1 states that the head change in the aquifer at the face of the well is equal to that inside the well; one assumes that there is no exit loss at the well face. Equation 2 states that the head change approaches zero as distance from the discharging well approaches infinity, a condition which will be approximated if boundaries of the aquifer are sufficiently distant from the discharging well. Equation 3 states that near the well the radial flow is equal to the rate of change in volume of water inside the well. Equations 4 and 5 state that initially the head change is zero in the aquifer, and the head increase or decrease inside the well is equal to H_0.

Solution (Cooper and others, 1967):

$$h = (2H_0/\pi)\int_0^\infty (\exp(-\beta u^2/\alpha)\{J_0(ur/r_w)\cdot[uY_0(u)-2\alpha Y_1(u)]-Y_0(ur/r_w)\cdot[uJ_0(u)-2\alpha J_1(u)]\}/\Delta(u))\,du, \quad (6)$$

where $\alpha = r_w^2 S/r_c^2$,

$\beta = Tt/r_c^2$,

and $\Delta(u) = [uJ_0(u)-2\alpha J_1(u)]^2 + [uY_0(u)-2\alpha Y_1(u)]^2$.

J_0 and Y_0, J_1 and Y_1, are zero-order and first-order Bessel functions of the first and second kind, respectively.

The head, H, inside the well, obtained by substituting $r=r_w$ in equation (6) is

$$H/H_0 = F(\beta,\alpha),$$

where

$$F(\beta,\alpha) = (8\alpha/\pi^2)\int_0^\infty (\exp(-\beta u^2/\alpha)/u\Delta(u))\,du$$

and where α, β, $\Delta(u)$ are as defined previously.

Comments: Figure 9.1 is a cross section showing geometric configuration along the well bore. The volume of water injected into or discharged from the well is $\pi r_c^2 H_0$. The water-level data in the injected well, expressed as a fraction of H_0, is plotted versus time on semilogarithmic graph paper. This plot is superimposed on figure 9.2, keeping the baselines the same and sliding horizontally until a match or interpolated fit is made. A match point for β, t, and α is picked from the two graphs. Transmissivity is calculated from $T=\beta r_c^2/t$ and storage coefficient from $S = \alpha r_c^2/r_w^2$. As pointed out by Cooper, Bredehoeft, and Papadopulos (1967, p. 267), the determination of S by this method has questionable reliability because of the similar shape of the curves, whereas the determination of T is not as sensitive to choosing the correct curve. Figure 9.2 on plate 1 is plotted from data in table 9.1, which contains original material from two sources (Cooper and others, 1967; and Papadopulos and others, 1973).

Table 9.2 is a listing of a FORTRAN program by S. S. Papadopulos that evaluates $F(\beta,\alpha)$. Input to the program consists of cards coded in a specific format (readers unfamiliar with FORTRAN formats should refer to a FORTRAN language manual). Input consists of two or more cards, each containing a single value of

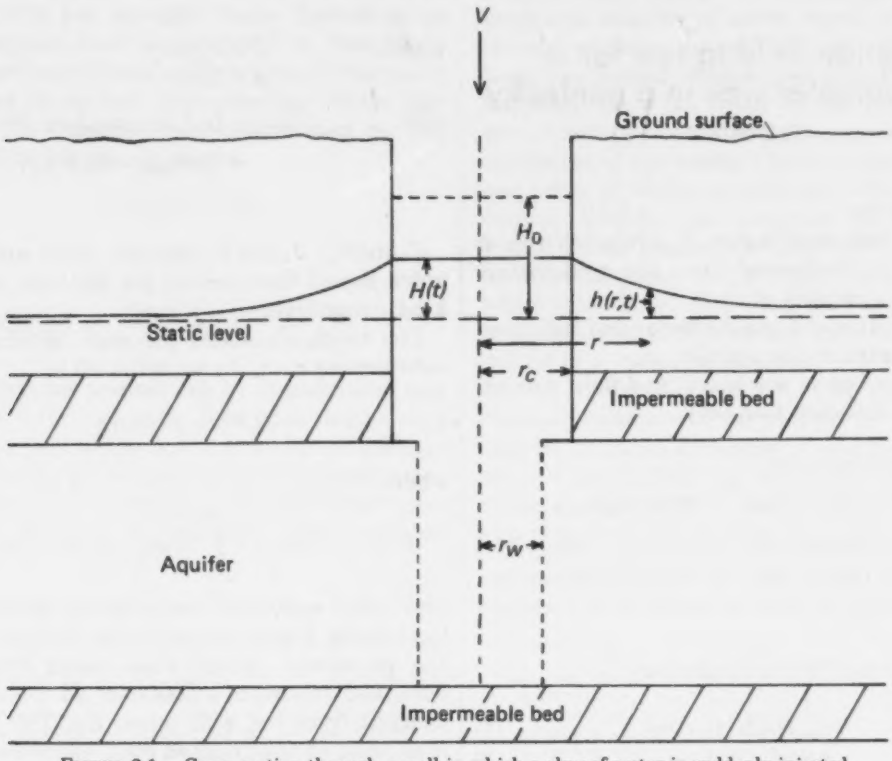

FIGURE 9.1.—Cross section through a well in which a slug of water is suddenly injected.

α coded in format F16.5. The first $\alpha \leq 0$ will signal program termination. Output from the program is shown in figure 9.3.

Solution 10: Constant discharge from a fully penetrating well in an aquifer that is anisotropic in the horizontal plane

Assumptions:

1. Well discharges at a constant rate, Q.
2. Well is of infinitesimal diameter and fully penetrates the aquifer.
3. Aquifer is anisotropic in the horizontal plane.
4. Aquifer is not leaky.
5. The transmissivity of the aquifer, T, is a two-dimensional symmetric tensor.

Differential equation:

$$T_{xx}\,\partial^2 s/\partial x^2 + 2T_{xy}\partial^2 s/\partial x \partial y + T_{yy}\partial^2 s/\partial y^2 + Q\,\delta(x)\delta(y) = S\partial s/\partial t.$$

This differential equation describes nonsteady flow in a homogeneous anisotropic aquifer with a constantly discharging well at $x=y=0$. The Dirac delta function is represented as $\delta(z)$ and has the following properties: $\delta(z)=0$ if $z \neq 0$ and $\int_{-\infty}^{+\infty} \delta(z)dz = 1$.

Boundary and initial conditions:

$$s(x,y,0)=0 \quad (1)$$
$$s(\pm\infty,y,t)=0 \quad (2)$$
$$s(x,\pm\infty,t)=0 \quad (3)$$

TABLE 9.1.—*Values of* H/H_0

From Cooper, Bredehoeft, and Papadopulos, 1967

Tt/r_c^2	α	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
10^{-3}	1.00	0.9771	0.9920	0.9969	0.9985	0.9992
	2.15	.9658	.9876	.9949	.9974	.9985
	4.64	.9490	.9807	.9914	.9954	.9970
10^{-2}	1.00	.9238	.9693	.9853	.9915	.9942
	2.15	.8860	.9505	.9744	.9841	.9883
	4.64	.8293	.9187	.9545	.9701	.9781
10^{-1}	1.00	.7460	.8655	.9183	.9434	.9572
	2.15	.6289	.7782	.8538	.8935	.9167
	4.64	.4782	.6436	.7436	.8031	.8410
10^{0}	1.00	.3117	.4598	.5729	.6520	.7080
	2.15	.1665	.2597	.3543	.4364	.5038
	4.64	.07415	.1086	.1554	.2082	.2620
	7.00	.04625	.06204	.08519	.1161	.1521
10^{1}	1.00	.03065	.03780	.04821	.06355	.08378
	1.40	.02092	.02414	.02844	.03492	.04426
	2.15	.01297	.01414	.01545	.01723	.01999
	3.00	.009070	.009615	.01016	.01083	.01169
	4.64	.005711	.004919	.006111	.006319	.006554
	7.00	.003722	.003809	.003884	.003962	.004046
10^{2}	1.00	.002577	.002618	.002653	.002688	.002725
	2.15	.001179	.001187	.001194	.001201	.001208

From Papadopulos, Bredehoeft, and Cooper, 1973

Tt/r_c^2	α	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-10}
10^{-3}	1	0.9994	0.9996	0.9996	0.9997	0.9997
	2	.9989	.9992	.9993	.9994	.9995
	4	.9980	.9985	.9987	.9989	.9991
	6	.9972	.9978	.9982	.9984	.9986
	8	.9964	.9971	.9976	.9980	.9982
10^{-2}	1	.9956	.9965	.9971	.9975	.9978
	2	.9919	.9934	.9944	.9952	.9958
	4	.9848	.9875	.9894	.9908	.9919
	6	.9782	.9819	.9846	.9866	.9881
	8	.9718	.9765	.9799	.9824	.9844
10^{-1}	1	.9655	.9712	.9753	.9784	.9807
	2	.9361	.9459	.9532	.9587	.9631
	4	.8828	.8995	.9122	.9220	.9298
	6	.8345	.8569	.8741	.8875	.8984
	8	.7901	.8173	.8383	.8550	.8686
10^{0}	1	.7489	.7801	.8045	.8240	.8401
	2	.5800	.6235	.6591	.6889	.7139
	3	.4554	.5033	.5442	.5792	.6096
	4	.3613	.4093	.4517	.4891	.5222
	5	.2893	.3351	.3768	.4146	.4487
	6	.2337	.2759	.3157	.3525	.3865
	7	.1903	.2285	.2655	.3007	.3337
	8	.1562	.1903	.2243	.2573	.2888
	9	.1292	.1594	.1902	.2208	.2505
10^{1}	1	.1078	.1343	.1620	.1900	.2178
	2	.02720	.03343	.04129	.05071	.06149
	3	.01286	.01448	.01667	.01956	.02320
	4	.008337	.008898	.009637	.01062	.01190
	5	.006209	.006470	.006789	.007192	.007709
	6	.004961	.005111	.005283	.005487	.005735
	8	.003547	.003617	.003691	.003773	.003863
10^{2}	1	.002763	.002803	.002845	.002890	.002938
	2	.001313	.001322	.001330	.001339	.001348

F(BETA,ALPHA) FOR ALPHA= 1.00D-01

BETA	H/H0
1.00D-03	0.9769
2.00D-03	0.9670
4.00D-03	0.9528
6.00D-03	0.9417
8.00D-03	0.9322
1.00D-02	0.9238
2.00D-02	0.8904
4.00D-02	0.8421
6.00D-02	0.8048
8.00D-02	0.7734
1.00D-01	0.7459
2.00D-01	0.6418
4.00D-01	0.5095
6.00D-01	0.4227
8.00D-01	0.3598
1.00D 00	0.3117
2.00D 00	0.1786
3.00D 00	0.1196
4.00D 00	0.0876
5.00D 00	0.0681
6.00D 00	0.0553
7.00D 00	0.0463
8.00D 00	0.0396
9.00D 00	0.0346
1.00D 01	0.0306
2.00D 01	0.0141
3.00D 01	0.0091
4.00D 01	0.0067
5.00D 01	0.0053
6.00D 01	0.0044
7.00D 01	0.0037
8.00D 01	0.0032
9.00D 01	0.0029
1.00D 02	0.0026
2.00D 02	0.0013
4.00D 02	0.0006
6.00D 02	0.0004
8.00D 02	0.0003
1.00D 03	0.0003

FIGURE 9.3.—Example of output from program to compute change in water level due to sudden injection of a slug of water into a well.

Equation 1 states that, initially, drawdown is zero. Equations 2 and 3 state that the drawdown approaches zero as distance from the discharging well approaches infinity, a condition which will be approximated if boundaries of the aquifer are sufficiently distant from the discharging well.

Solution (Papadopulos, 1965, p. 23):

$$s = (Q/4\pi\sqrt{T_{xx}T_{yy}-T_{xy}^2})\, W(u_{xy}), \qquad (4)$$

where

$$W(u) = \int_u^\infty (e^{-v}/v)\, dv$$

and

$$u_{xy} = (S/4t)(T_{xx}y^2 + T_{yy}x^2 - 2T_{xy}xy)/(T_{xx}T_{yy} - T_{xy}^2). \qquad (5)$$

If the coordinate axes x and y are the same as the principal axes ϵ and η (fig. 10.1) of the transmissivity tensor, the preceding equation for drawdown becomes

$$s = (Q/4\pi\sqrt{T_{\epsilon\epsilon}\ T_{\eta\eta}})\, W(u_{\epsilon\eta}),$$

where

$$u_{\epsilon\eta} = (S/4t)(T_{\epsilon\epsilon}\ n^2 + T_{\eta\eta}\ \epsilon^2)/T_{\epsilon\epsilon}\ T_{\eta\eta}.$$

Comments: The method of type-curve solution as outlined by Papadopulos (1965, p. 26) requires observation of drawdown in at least three observation wells. First, choose a convenient rectangular coordinate system with the pumped well at the origin. Then, plot the observed drawdown versus t on logarithmic paper. Match these plots to the $W(u)$ type curve given in solution 1. Choose a match point of (t,s) and $(1/u_{xy}, W(u_{xy}))$ for each well and compute $T_{xx}T_{yy}-T_{xy}^2 = (QW(u_{xy})/4\pi s)^2$ for each well. Match points for all observation wells should yield approximately the same value of $(T_{xx}T_{yy}-T_{xy}^2)$. Usually they will not and judgment must be used to obtain an "average" value. Substituting this value and the three values of (x,y) in equation 5 gives three equations in three unknowns ST_{xx}, ST_{yy}, and ST_{xy}. These equations are of the form

$$y^2(ST_{xx}) + x^2(ST_{yy}) - 2xy(ST_{xy}) = 4tu_{xy}(T_{xx}T_{yy} - T_{xy}^2).$$

Solve these three equations to determine T_{xx}, T_{xy}, and T_{yy} in terms of S, and S may be determined from

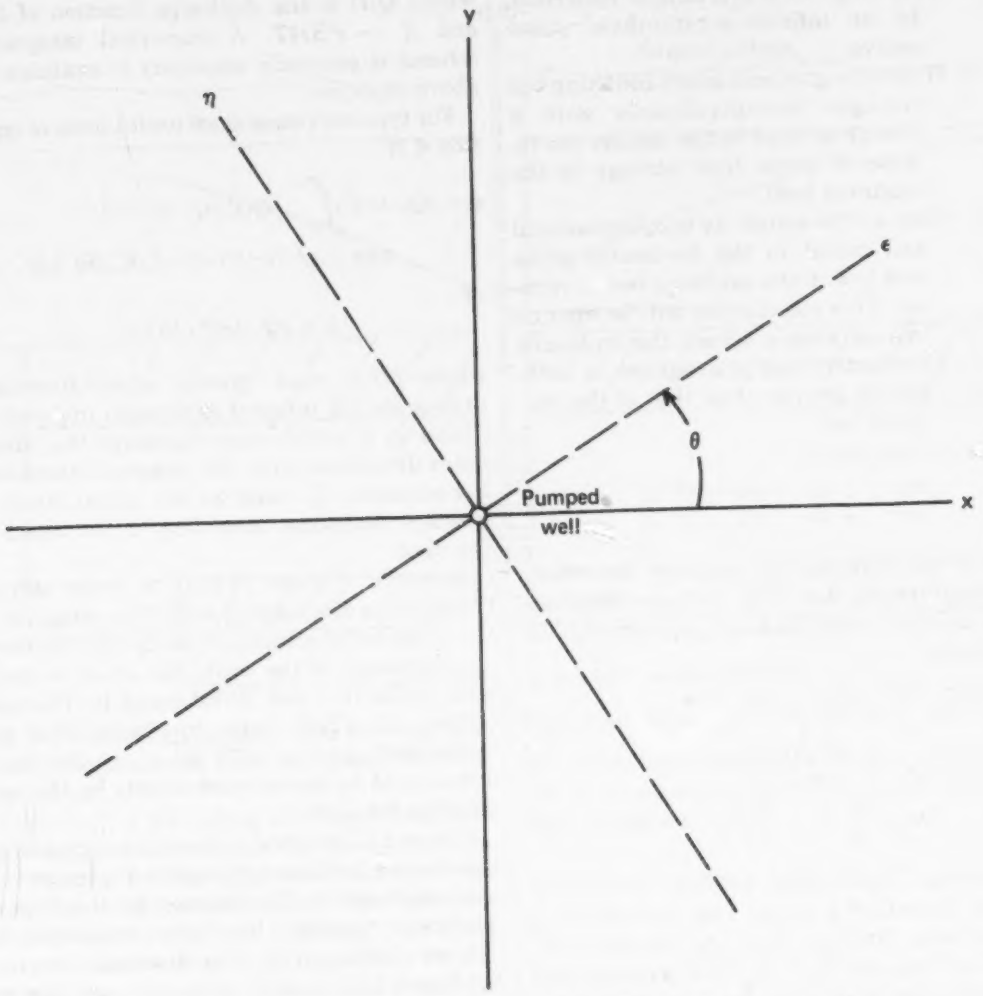

FIGURE 10.1.—Plan view showing coordinate axes.

$$S = \sqrt{(ST_{xx}ST_{yy} - (ST_{xy})^2)/(T_{xx}T_{yy} - T_{xy}^2)}.$$

Then, compute T_{xx}, T_{yy}, and T_{xy} from ST_{xx}, ST_{yy}, and ST_{xy}. $T_{\epsilon\epsilon}$, $T_{\eta\eta}$, and Θ (the angle between the x and the ϵ axis) may be calculated from the relations (Papadopulos, 1965, p. 28)

$$T_{\epsilon\epsilon} = 1/2(T_{xx} + T_{yy} + ((T_{xx} - T_{yy})^2 + 4T_{xy}^2)^{1/2})$$

$$T_{\eta\eta} = 1/2(T_{xx} + T_{yy} - ((T_{xx} - T_{yy})^2 + 4T_{xy}^2)^{1/2})$$

$$\Theta = \arctan((T_{\epsilon\epsilon} - T_{xx})/T_{xy}).$$

Solution 11: Variable discharge from a fully penetrating well in a leaky aquifer

Assumptions:

1. Well discharge changes as a specified function of time.
2. Well is of infinitesimal diameter and fully penetrates the aquifer.
3. Aquifer is overlain, or underlain, everywhere by a confining bed having uniform hydraulic conductivity (K') and thickness (b').

4. Confining bed is overlain, or underlain, by an infinite constanthead plane source.
5. Hydraulic gradient across confining bed changes instantaneously with a change in head in the aquifer (no release of water from storage in the confining bed).
6. Flow in the aquifer is two-dimensional and radial in the horizontal plane and flow in the confining bed is vertical. This assumption will be approximated closely where the hydraulic conductivity of the aquifer is sufficiently greater than that of the confining bed.

Differential equation:

$$\frac{\partial^2 s}{\partial r^2} + \frac{1}{r}\frac{\partial s}{\partial r} - \frac{sK'}{Tb'} = \frac{S}{T}\frac{\partial s}{\partial t}$$

This is the differential equation describing nonsteady radial flow in a homogeneous isotropic aquifer with leakage proportional to drawdown.

Boundary and initial conditions:

$$s(r,0)=0 \quad (1)$$

$$s(\infty,t)=0 \quad (2)$$

$$\lim_{r\to 0} r\frac{\partial s}{\partial r} = -\frac{Q(t)}{2\pi T}, \; t \geq 0 \quad (3)$$

Equation 1 states that, initially, drawdown is zero. Equation 2 states that drawdown is zero at large distances from the pumped well. Equation 3 states that near the pumped well the radial flow is equal to the discharge of the pumped well, which is a function of time.

Solution:

Solutions for certain discharge functions have been published by Abu-Zied and Scott (1963), and Werner (1946) for a nonleaky aquifer, and by Hantush (1964a) for both leaky and nonleaky aquifers. For arbitrary discharge functions for leaky aquifers, a solution using the convolution integral has been presented by Moench (1971, eq. 3):

$$s = (1/4\pi T)\int_0^t (Q(t')/(t-t')) \cdot \exp\left(-A/(t-t') - (t-t')K'/Sb'\right)dt', \quad (4)$$

where $Q(t)$ is the discharge function of time and $A = r^2S/4T$. A numerical integration scheme is generally necessary to evaluate the above equation.

For type curves, a more useful form of equation 4 is

$$s = (Q_r/4\pi T)\int_0^t \left[Q(t')/Q_r\,(t-t')\right] \cdot \exp\left[-A/(t-t') - (t-t')K'/Sb'\right]dt', \quad (5)$$

or

$$s = (Q_r/4\pi T)\,SO(t), \quad (6)$$

where $SO(t)$, read "system output function," represents the integral expression in equation 5, and Q_r is an arbitrary discharge that eliminates dimension from the integral expression. For example, Q_r could be the initial, final, or average discharge, according to the needs of the user.

Comments: Figure 11.1 is a cross section through the discharging well. This situation is the same as for solution 4, except for the varying discharge of the well. The effect of finite well radius (r_w) was investigated by Hantush (1964b, p. 4224), who concluded that for $t > 25r_w^2S/T$ and $r_w/\sqrt{Tb'/K'} < 0.1$ the drawdown could be represented closely by the convolution integral.

Figure 11.2 on plate 1 shows a selected set of type curves for linear change in discharge in a nonleaky aquifer. The solution for this type of discharge function has been presented by Werner (1946, p. 706). The discharge function for figure 12.2 is $Q(t)=Q_0(1+ct)$, and the resulting drawdown is

$$s = (Q_0/4\pi T)W(u)\left\{1+ct\left[u+1-e^{-u}/W(u)\right]\right\},$$

where $W(u)$ is the well function of Theis. Substituting A/u for t in the above expression gives

$$s = (Q_0/4\pi T)\,W(u) \cdot (1+cA\left\{1+(1/u)\left[1-e^{-u}/W(u)\right]\right\}),$$

or

$$s = (Q_0/4\pi T)\,SO(t),$$

where $SO(t)$ represents

$$W(u)\left(1+cA\left\{1+(1/u)\left[1-e^{-u}/W(u)\right]\right\}\right).$$

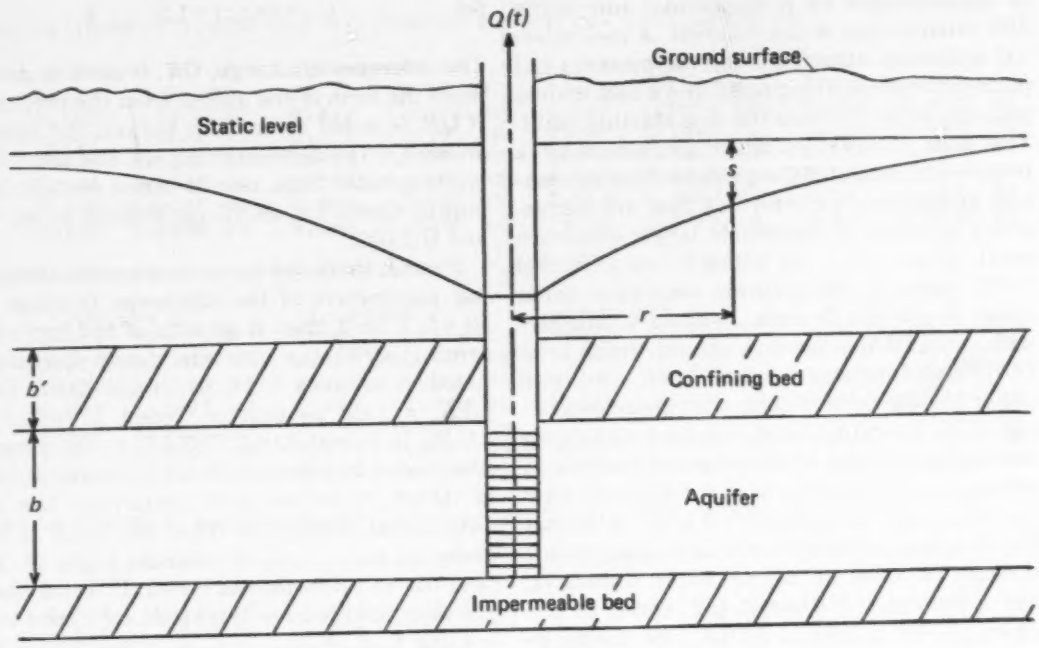

FIGURE 11.1.—Cross section through a well with variable discharge.

This substitution permits the plotting of a family of type curves, each curve specified by a value of cA.

Table 11.1 is the listing of a FORTRAN program designed to evaluate the above convolution integral for five different discharge functions. Three of these discharge functions are those devised by Hantush (1964a, p. 343, 344), who presented solutions for drawdown resulting from these functions. These three discharge functions are:

(a) $Q(t)=Q_s\left[1+\delta \exp\left(-t/t^*\right)\right]$,

(b) $Q(t)=Q_s\left[1+\delta/(1+t/t^*)\right]$,

and (c) $Q(t)=Q_s\left[1+\delta/\sqrt{1+t/t^*}\right]$,

where Q_s is the ultimate steady discharge and δ and t^* are parameters defining a particular function. The first discharge function, for an exponentially decreasing discharge (case "a" of Hantush, 1964a) is virtually the same as the discharge function of Abu-Zied and Scott (1963). Besides the three functions of Hantush, the program also includes discharge as a fifth-degree polynomial of time, $Q(t)=\sum_{i=0}^{5} a_i t^i$, where the a_i are the coefficients of the polynomial, and as a piecewise linear function of time with eight segments,

$$Q(t)=a_j+b_j(t-t_{j-1})$$

for

$$t_{j-1}<t\leq t_j, j=1, 2, \ldots, 8,$$

where a_j and b_j are parameters defining the j^{th} line segment. The program uses a different, but equivalent to equation 4, expression for the convolution integral

$$s = (1/4\pi T)\int_0^t (Q(t-t')/t') \cdot \exp\left(-A/t'-t'K'/Sb'\right) dt'.$$

The program uses a sum to approximate the convolution integral. It chooses a starting value of t' that satisfies $r^2S/4Tt' + K't'/Sb' = 100$. If such a value of t' does not exist, that is, $(r^2S/4T)(K'/Sb')>2500$, then a value of zero is assigned for the integral value. The ending point of the interval is picked as 10 times the

starting point. The integral over this interval is approximated by a trapezoidal sum using 500 subdivisions of the interval. A new interval is then constructed using the previous end point as a new starting point and a new ending point equal to 10 times the new starting point. This new interval is again evaluated by a trapezoidal sum of 500 segments. This summation procedure over intervals that are successively an order of magnitude larger continues until either $t'=t$ or $(r^2S/4Tt') + (K't/Sb') > 101$. Input to this program consists of cards coded in specific formats. Readers unfamiliar with FORTRAN formats should refer to a FORTRAN language manual. Input consists of one or more groups of data, each group consisting of the following. First, one card containing the beginning time of the period of analysis in columns 1–10, coded in format E10.3; the ending time coded in columns 1311–20, in format E10.3; and a discharge index (a number from 1 through 5) coded in column 25, in format I1; and a reference discharge, QR, coded in columns 31–40, in format E10.3. The discharge index, IQ, selects a discharge function, $Q(t)$, in the following manner. If $IQ=1$, the discharge function is exponentially decreasing,

$$Q(t)=Q_s\left[1+\delta\exp(-t/t^*)\right].$$

This is case (a) of Hantush (1964a, p. 343). If $IQ=2$, the discharge function is hyperbolically decreasing,

$$Q(t)=Q_s\left[1+\delta/(1+t/t^*)\right].$$

This is case (b) of Hantush (1964a, p. 344). If $IQ=3$, the discharge function is the same as case (c) of Hantush (1964a, p. 344),

$$Q(t)=Q_s\left[1+\delta/\sqrt{1+t/t^*}\right].$$

If $IQ=4$, the discharge function is a fifth-degree polynomial of time,

$$Q(t) = \sum_{i=0}^{5} a_i t^i.$$

If $IQ=5$, the discharge function is a piecewise-linear function of time with eight or less segments,

$$Q(t)=a_j+b_j(t-t_{j-1})$$
$$\text{for} \quad t_{j-1}<t\leq t_j, j=1,2,\ldots,8.$$

The reference discharge, QR, is used to determine the form of the output from the program: If QR is coded as zero (or blank), the output shows t, s (as defined by eq. 4), and $Q(t)$. If a value greater than zero is coded for QR, the output shows $1/u$, $SO(t)$ (as defined by eq. 6), and $Q(t)/QR$.

Second, there are one or more cards containing parameters of the discharge function. If $IQ=1$, 2, or 3, then it consists of one card containing: QST, the ultimate steady discharge, coded in columns 1–10, in format E10.3; DELTA, a rate parameter, coded in columns 11–20, in format E10.3; TSTAR, a time parameter, coded in columns 21–30, in format E10.3. If $IQ=4$, it is one card containing the six polynomial coefficients. They are coded in the order $a_0, a_1, \ldots, a_5$, in columns 1–10; 11–20, . . . , 51–60 all in format E10.3. If $IQ=5$, then the program requires four cards, each card containing t_j, a_j, b_j, t_{j+1}, a_{j+1}, b_{j+1}; the four cards representing $j=1, 3, 5, 7$. The last part of each set of data consists of two or more cards containing coded values for: distance from pumped well, in columns 1–10; storage coefficient, in columns 11–20; transmissivity, in columns 21–30; and ratio of hydraulic conductivity to thickness for the confining bed, in columns 31–40, all in format E10.3. A blank card is used to signal the end of each set of data. Output from this program is shown in figure 11.3.

References

Abu-Zied, M. A., and Scott, V. H., 1963, Nonsteady flow for wells with decreasing discharge: Hydraulics Div. Jour., Am. Soc. Civil Engineers Proc., v. 89, no. H–13, p. 119–132.

Cooper, H. H., Jr., 1963, Type curves for nonsteady radial flow in an infinite leaky artesian aquifer, *in* Bentall, Ray, compiler, Shortcuts and special problems in aquifer tests: U.S. Geol. Survey Water-Supply Paper 1545–C, p. C48–C55.

Cooper, H. H., Jr., Bredehoeft, J. D., and Papadopulos, I. S., 1967, Response of a finite-diameter well to an instantaneous charge of water: Water Resources Research, v. 3, no. 1, p. 263–269.

Dudley, W. W., Jr., 1970, Nonsteady inflow to a chamber

R**2*S/(4*TRANS)= 1.000E-04, K'/(S*B')= 2.500E 03, QR= 1.257E 05

1/U	1/U*10** 0		1/U*10** 1		1/U*10** 2		1/U*10** 3	
	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR
1.0	0.185	1.000E 00	0.819	1.000E 00	0.842	1.000E 00	0.842	1.000E 00
1.5	0.317	1.000E 00	0.837	1.000E 00	0.842	1.000E 00	0.842	1.000E 00
2.0	0.421	1.000E 00	0.841	1.000E 00	0.842	1.000E 00	0.842	1.000E 00
3.0	0.566	1.000E 00	0.842	1.000E 00	0.842	1.000E 00	0.842	1.000E 00
5.0	0.715	1.000E 00	0.842	1.000E 00	0.842	1.000E 00	0.842	1.000E 00
7.0	0.780	1.000E 00	0.842	1.000E 00	0.842	1.000E 00	0.842	1.000E 00

R**2*S/(4*TRANS)= 1.000E-04, K'/(S*B')= 2.500E 01, QR= 1.257E 05

1/U	1/U*10** 0		1/U*10** 1		1/U*10** 2		1/U*10** 3	
	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR
1.0	0.219	1.000E 00	1.805	1.000E 00	3.815	1.000E 00	4.829	1.000E 00
1.5	0.397	1.000E 00	2.167	1.000E 00	4.111	1.000E 00	4.849	1.000E 00
2.0	0.558	1.000E 00	2.427	1.000E 00	4.296	1.000E 00	4.853	1.000E 00
3.0	0.826	1.000E 00	2.793	1.000E 00	4.515	1.000E 00	4.854	1.000E 00
5.0	1.216	1.000E 00	3.244	1.000E 00	4.708	1.000E 00	4.854	1.000E 00
7.0	1.495	1.000E 00	3.530	1.000E 00	4.785	1.000E 00	4.854	1.000E 00

R**2*S/(4*TRANS)= 1.000E-04, K'/(S*B')= 2.500E-01, QR= 1.257E 05

1/U	1/U*10** 0		1/U*10** 1		1/U*10** 2		1/U*10** 3	
	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR
1.0	0.219	1.000E 00	1.823	1.000E 00	4.036	1.000E 00	6.307	1.000E 00
1.5	0.398	1.000E 00	2.196	1.000E 00	4.437	1.000E 00	6.700	1.000E 00
2.0	0.560	1.000E 00	2.468	1.000E 00	4.721	1.000E 00	6.975	1.000E 00
3.0	0.829	1.000E 00	2.857	1.000E 00	5.123	1.000E 00	7.356	1.000E 00
5.0	1.223	1.000E 00	3.354	1.000E 00	5.627	1.000E 00	7.820	1.000E 00
7.0	1.507	1.000E 00	3.684	1.000E 00	5.958	1.000E 00	8.110	1.000E 00

R**2*S/(4*TRANS)= 1.000E-04, K'/(S*B')= 2.500E-03, QR= 1.257E 05

1/U	1/U*10** 0		1/U*10** 1		1/U*10** 2		1/U*10** 3	
	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR	SO(T)	Q(T)/QR
1.0	0.219	1.000E 00	1.823	1.000E 00	4.038	1.000E 00	6.332	1.000E 00
1.5	0.398	1.000E 00	2.197	1.000E 00	4.440	1.000E 00	6.737	1.000E 00
2.0	0.560	1.000E 00	2.468	1.000E 00	4.726	1.000E 00	7.024	1.000E 00
3.0	0.829	1.000E 00	2.857	1.000E 00	5.130	1.000E 00	7.429	1.000E 00
5.0	1.223	1.000E 00	3.355	1.000E 00	5.639	1.000E 00	7.939	1.000E 00
7.0	1.507	1.000E 00	3.686	1.000E 00	5.975	1.000E 00	8.275	1.000E 00

FIGURE 11.3.—Example of output from program to compute the convolution integral for a leaky aquifer.

within a thick aquitard, *in* Geological Survey research 1970: U.S. Geol. Survey Prof. Paper 700–C, p. C206–C211.

Ferris, J. G., Knowles, D. B., Brown, R. H., and Stallman, R. W., 1962, Theory of aquifer tests: U.S. Geol. Survey Water-Supply Paper 1536–E, p. E69–E174.

Hantush, M. S., 1956, Analysis of data from pumping tests in leaky aquifers: Am. Geophys. Union Trans., v. 37, no. 6, p. 702–714.

———1959, Nonsteady flow to flowing wells in leaky aquifers: Jour. Geophys. Research, v. 64, no. 8, p. 1043–1052.

———1960, Modification of the theory of leaky aquifers: Jour. Geophys. Research, v. 65, no. 11, p. 3713–3725.

———1961a, Drawdown around a partially penetrating well: Hydraulics Div. Jour., Am. Soc. Civil Engineers Proc., p. 83–98.

———1961b, Aquifer tests on partially penetrating wells: Hydraulics Div. Jour., Am. Soc. Civil Engineers Proc., p. 171–195.

———1961c, Tables of the function $M(u,\beta)$: New Mexico Inst. Mining and Technology Prof. Paper 102, 34 p.

———1961d, Tables of the function $H(u,\beta)=\int_0^u \frac{e^{-y}}{y}\,\mathrm{erfc}\left[\frac{\beta\sqrt{u}}{\sqrt{y(y-u)}}\right]dy$: New Mexico Inst. Mining and Technology Prof. Paper 103, 12 p.

———1961e, Tables of the function $W(u,\beta)=\int_0^u \frac{e^{-y-\frac{\beta^2}{4y}}}{y}\,dy$: New Mexico Inst. Mining and Technology Prof. Paper 104, 13 p.

———1962, Flow of ground water in sands of nonuniform thickness, 3,Flow to wells: Jour. Geophys. Research, v. 67, no. 4, p. 1527–1534.

———1964a, Hydraulics of wells, *in* Chow, Ven Te, ed., Advances in hydroscience, Volume 1: New York, Academic Press, p. 281–442.

———1964b, Drawdown around wells of variable discharge: Jour. Geophys. Research, v. 69, no. 20, p. 4221–4235.

———1967, Flow of groundwater in relatively thick leaky aquifers: Water Resources Research, v. 3, no. 2, p. 583–590.

Hantush, M. S., and Jacob, C. E., 1954, Plane potential flow of ground water with linear leakage: Am. Geophys. Union Trans., v. 35, no. 6, p. 917–936.

———1955, Non-steady radial flow in an infinite leaky aquifer: Am. Geophys. Union Trans., v. 36, no. 1, p. 95–100.

Jacob, C. E., 1946, Radial flow in a leaky artesian aquifer: Am. Geophys. Union Trans., v. 27, no. 2, p. 198–205.

Jacob, C. E., and Lohman, S. W., 1952, Nonsteady flow to a well of constant drawdown in an extensive aquifer: Am. Geophys. Union Trans., v. 33, no. 4, p. 559–569.

Lohman, S. W., 1972, Ground-water hydraulics: U.S. Geol. Survey Prof. Paper 708, 70 p.

Lohman, S. W., and others, 1972, Definitions of selected ground-water terms—revisions and conceptual refinements: U.S. Geol. Survey Water-Supply Paper 1988, 21 p.

Moench, Allen, 1971, Ground-water fluctuations in response to arbitrary pumpage: Ground Water, v. 9, no. 2, p. 4–8.

Neuman, S. P., and Witherspoon, P. A., 1969a, Theory of flow in a confined two aquifer system: Water Resources Research, v. 5, no. 4, p. 803–816.

———1969b, Applicability of current theories of flow in leaky aquifers: Water Resources Research, v. 5, no. 4, p. 817–829.

———1971, Transient flow of ground water to wells in multiple-aquifer systems, *in* Sea-water intrusion: Aquitards in the coastal ground water basin of Oxnard Plain, Ventura County: California Department of Water Resources Bulletin 63–4, p. 159–359.

Papadopulos, I. S., 1965, Nonsteady flow to a well in an infinite anisotropic aquifer: Internat. Assoc. Sci. Hydrology Symposium, Dubrovnik [Yugoslavia], 1965, p. 21–31.

———1967, Drawdown distribution around a large-diameter well: Natl. Symposium on Ground-Water Hydrology, San Francisco, Calif., Nov. 6–8, 1967, Proc., p. 157–168.

Papadopulos, I. S., and Cooper, H. H., Jr., 1967, Drawdown in a well of large diameter: Water Resources Research, v. 3, no. 1, p. 241–244.

Papadopulos, S. S., Bredehoeft, J. D., and Cooper, H. H., Jr., 1973, On the analysis of slug test data: Water Resources Research, v. 9, no. 4, p. 1087–1089.

Rorabaugh, M. I., 1953, Graphical and theoretical analysis of step-drawdown test of artesian wells: Am. Soc. Civil Engineers Trans., v. 79, Separate 362, 23 p.

Stallman, R. W., 1971, Aquifer-test design, observation, and data analysis: U.S. Geol. Survey Techniques Water-Resources Inv., book 3, chap. B1, 26 p.

Theis, C. V., 1935, The relation between the lowering of the piezometric surface and the rate and duration of discharge of a well using ground-water storage: Am. Geophys. Union Trans., v. 16, p. 519–524.

Walton, W. C., 1962, Selected analytical methods for well and aquifer evaluation: Illinois State Water Survey Bull. 49, 81 p.

Weeks, E. P., 1964, Field methods for determining vertical permeability and aquifer anisotropy: U.S. Geol. Survey Prof. Paper 501-D, p. D193–D198.

———1969, Determining the ratio of horizontal to vertical permeability by aquifer-test analysis: Water Resources Research, v. 5, no. 1, p. 196–214.

Wenzel, L. K., 1942, Methods for determining permeability of water-bearing materials, with special reference to discharging-well methods: U.S. Geol. Survey Water-Supply Paper 887, 192 p.

Werner, P. W., 1946, Notes on flow-time effects in the great artesian aquifers of the earth: Am. Geophys. Union Trans., v. 27, no. 5, p. 687–708.

SUPPLEMENTAL DATA

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*

```
C     ************************************************************************PPN   1
C                                                                               PPN   2
C                                                                               PPN   3
C     PURPOSE                                                                   PPN   4
C        TO COMPUTE TYPE CURVE FUNCTION VALUES FOR PARTIAL PENETRATION          PPN   5
C        IN A NONLEAKY AQUIFER USING EQUATIONS 1 AND 9A OF HANTUSH,M.S.,        PPN   6
C        1961,DRAWDOWN AROUND A PARTIALLY PENETRATING WELL: HYDRAULIC           PPN   7
C        DIV. JOUR.,, AM. SOC. CIVIL ENGINEERS PROC.,, P. 83-98.                PPN   8
C     INPUT DATA                                                                PPN   9
C        1 CARD  =  FORMAT (3F5.1,I5,2E10.4)                                    PPN  10
C              B = AQUIFER THICKNESS                                            PPN  11
C              L = DEPTH, BELOW TOP OF AQUIFER, TO BOTTOM OF PUMPING            PPN  12
C              WELL SCREEN                                                      PPN  13
C              D = DEPTH, BELOW TOP OF AQUIFER, TO TOP OF PUMPING WELL          PPN  14
C              SCREEN                                                           PPN  15
C              NUM = NUMBER OF OBSERVATION WELLS OR PIEZOMETERS TIMES           PPN  16
C              NUMBER OF VALUES OF KZ/KR,                                       PPN  17
C              SMALL = SMALLEST VALUE OF 1/U FOR WHICH COMPUTATION IS           PPN  18
C              DESIRED                                                          PPN  19
C              LARGE = LARGEST  VALUE OF 1/U FOR WHICH COMPUTATION IS           PPN  20
C              DESIRED                                                          PPN  21
C        NUM CARDS (ONE FOR EACH OBS. WELL OR PIEZOMETER AND FOR EACH           PPN  22
C              VALUE OF R*SQRT(KZ/KR), = FORMAT (3F5.1)                         PPN  23
C              R = RADIAL DISTANCE FROM PUMPED WELL TIMES SQRT(KZ/KR),          PPN  24
C              LPRIME = DEPTH, BELOW TOP OF AQUIFER, TO BOTTOM OF OBS.          PPN  25
C              WELL SCREEN (ZERO FOR PIEZOMETER)                                PPN  26
C              DPRIME = DEPTH, BELOW TOP OF AQUIFER, TO TOP OF OBS. WELL        PPN  27
C              SCREEN (TOTAL DEPTH FOR PIEZOMETER)                              PPN  28
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                             PPN  29
C        DQL12,SERIES,BESK,FCT,L,F,EXPI                                         PPN  30
C     ************************************************************************PPN  31
      REAL*8 U                                                                  PPN  32
      REAL*4 L,LB,LPB,LPRIME,LARGE                                              PPN  33
      DIMENSION ARRAY(13,12), IARG(12), ARG(13), A(12), C(12)                   PPN  34
      DATA ARG/1.,1.2,1.5,2.,2.5,3.,3.5,4.,5.,6.,7.,8.,9./                      PPN  35
      DATA A/12*1.  N*!/,C/12*1.0**1/                                           PPN  36
      IRD=5                                                                     PPN  37
      IPT=6                                                                     PPN  38
      READ (IRD,6) B,L,D,NUM,SMALL,LARGE                                        PPN  39
      LB=L/B                                                                    PPN  40
      DB=D/B                                                                    PPN  41
      IBEGIN=ALOG10(SMALL)                                                      PPN  42
      IEND=ALOG10(LARGE)+1.                                                     PPN  43
      JLIMIT=IEND-IBEGIN                                                        PPN  44
      IF (JLIMIT.GT.12) JLIMIT=12                                               PPN  45
      DO 5 K=1,NUM                                                              PPN  46
      READ (IRD,6) R,LPRIME,DPRIME                                              PPN  47
      RB=R/B                                                                    PPN  48
      LPB=LPRIME/B                                                              PPN  49
      DPB=DPRIME/B                                                              PPN  50
      DO 1 I=1,13                                                               PPN  51
      ARGI=ARG(I)                                                               PPN  52
      DO 1 J=1,JLIMIT                                                           PPN  53
      IARG(J)=IBEGIN+J-1                                                        PPN  54
      Y=ARGI*10.**(IBEGIN+J-1)                                                  PPN  55
      U=1./Y                                                                    PPN  56
      X=U                                                                       PPN  57
      CALL EXPI(X,WU,DUMMY)                                                     PPN  58
    1 ARRAY(I,J)=WU*F(U,RB,LB,DB,LPB,DPB)                                       PPN  59
      IF (LPB=0.) 2,2,3                                                         PPN  60
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
      2 WRITE (IPT,7) DPB,RB,LB,DB                                          PPN   61
        GO TO 4                                                             PPN   62
      3 WRITE (IPT,8) LPB,DPB,RB,LB,DB                                      PPN   63
      4 WRITE (IPT,9) (A(I),C(I),IARG(I),I=1,JLIMIT)                        PPN   64
        DO 5 I=1,13                                                         PPN   65
        WRITE (IPT,10) ARG(I),(ARRAY(I,J),J=1,JLIMIT)                       PPN   66
      5 CONTINUE                                                            PPN   67
        STOP                                                                PPN   68
C                                                                           PPN   69
C                                                                           PPN   70
      6 FORMAT (3F5.1,I5,2E10.4)                                            PPN   71
      7 FORMAT ('1','W(U)=F(U,R/B,L/B,D/B,Z/B)', Z/B=',F5.2,', SQRT(KZ/KR)*PPN   72
     1R/B=',F5.2,', L/B=',F5.2,', D/B=',F5.2,', U=1/N')                     PPN   73
      8 FORMAT ('1','W(U)=F(U,R/B,L/B,D/B,L''/B,D''/B), L''/B=',F5.2,', D'PPN   74
     1'/B=',F5.2,', SQRT(KZ/KR)*R/B=',F5.2,', L/B=',F5.2,', D/B=',F5.2,'PPN   75
     2, U=1/N')                                                             PPN   76
      9 FORMAT ('0',2X,'N',1X,12(2A4,I2))                                   PPN   77
     10 FORMAT ((' ',F4.1,12(F9.4,1X)))                                     PPN   78
        END                                                                 PPN   79*
        REAL FUNCTION F*4(U,RB,LB,DB,LPB,DPB)                                F    1
C       ***************************************************************      F    2
C                                                                            F    3
C       FUNCTION F                                                           F    4
C                                                                            F    5
C       PURPOSE                                                              F    6
C          TO COMPUTE DEPARTURES FROM THEIS CURVE CAUSED BY PARTIAL          F    7
C          PENETRATION OF PUMPED WELL.                                       F    8
C       USAGE                                                                F    9
C          F(U,RB,LB,DB,LPB,DPB)                                             F   10
C       DESCRIPTION OF PARAMETERS                                            F   11
C          ALL REAL, U DOUBLE PRECISION                                      F   12
C          U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE            F   13
C              COEFFICIENT / 4*TRANSMISSIVITY * TIME                         F   14
C          RB = R/B ( RADIAL DISTANCE / AQUIFER THICKNESS )                  F   15
C          LB = L/B ( FRACTION OF AQUIFER PENETRATED BY PUMPED WELL)         F   16
C          DB = D/B ( FRACTION OF AQUIFER ABOVE PUMPED WELL SCREEN)          F   17
C          LPB = L'/B (FRACTION OF AQUIFER PENETRATED BY OBS, WELL, ZERO     F   18
C                FOR PIEZOMETER)                                             F   19
C          DPB = D'/B (FRACTION OF AQUIFER ABOVE OBS, WELL SCREEN, TOTAL     F   20
C                DEPTH FOR PIEZOMETER)                                       F   21
C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                        F   22
C          DQL12,SERIES,BESK,FCT,L                                           F   23
C       METHOD                                                               F   24
C          SUMS THE SERIES THROUGH N*PI*R/B EQ 20                            F   25
C                                                                            F   26
C       ***************************************************************      F   27
        REAL*8 U,V                                                           F   28
        REAL*4 L,N,LB,LPB                                                    F   29
        SUM=0.                                                               F   30
        N=0.                                                                 F   31
        PIRB=3.141593*RB                                                     F   32
        PILB=3.141593*LB                                                     F   33
        PIDB=3.141593*DB                                                     F   34
        IF (LPB=0.) 1,1,4                                                    F   35
C       CHECKS FOR WELL OR PIEZOMETER                                        F   36
      1 PIZB=3.141593*DPB                                                    F   37
      2 N=N+1.                                                               F   38
        V=N*PIRB/2.                                                          F   39
        IF (V.GT.10.) GO TO 3                                                F   40
C       TRUNCATES SERIES WHEN V>10                                           F   41
        X=L(U,V)/N                                                           F   42
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
      SUM=SUM+(SIN(N*PILB)-SIN(N*PIDB))*COS(N*PIZB)*X                    F  43
      GO TO 2                                                           F  44
    3 F=.6366198*SUM/(LB-DB)                                            F  45
      GO TO 7                                                           F  46
    4 PILPB=3.141593*LPB                                                F  47
      PIDPB=3.141593*DPB                                                F  48
    5 N=N+1                                                             F  49
      V=N*PIRB/2.                                                       F  50
      IF (V.GT.10.) GO TO 6                                             F  51
C     TRUNCATES SERIES WHEN V>10                                        F  52
      X=L(U,V)/N                                                        F  53
      SUM=SUM+(SIN(N*PILB)-SIN(N*PIDB))*(SIN(N*PILPB)-SIN(N*PIDPB))*X/N F  54
      GO TO 5                                                           F  55
    6 F=.2026424*SUM/((LB-DB)*(LPB-DPB))                                F  56
    7 RETURN                                                            F  57
      END                                                               F  58
      REAL FUNCTION L*8(U,V)                                            L   1
C     ***************************************************************** L   2
C                                                                       L   3
C     FUNCTION L                                                        L   4
C                                                                       L   5
C     PURPOSE                                                           L   6
C        TO COMPUTE THE INTEGRAL( EXP(-Y-V**2/Y)/Y) SUMMED OVER Y FROM  L   7
C        U TO INFINITY(WELL FUNCTION FOR LEAKY AQUIFERS).               L   8
C     DESCRIPTION OF PARAMETERS                                         L   9
C        BOTH DOUBLE PRECISION                                          L  10
C        U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE         L  11
C            COEFFICIENT / 4*TRANSMISSIVITY * TIME                      L  12
C        V = R/2*SQRT(K'/(T*B'))=ONE-HALF RADIAL DISTANCE*SQUARE ROOT   L  13
C            (HYD. COND. OF CONFINING BED/TRANSMISSIVITY*THICKNESS      L  14
C            OF CONFINING BED)                                          L  15
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                     L  16
C        DQL12,SERIES,BESK,FCT                                          L  17
C     METHOD                                                            L  18
C        IN THE FOLLOWING F=EXP(-Y-V**2/Y)/Y                            L  19
C       (1) U>=1, USES A GAUSSIAN-LAGUERRE QUADRATURE FORMULA TO        L  20
C         EVALUATE INTEGRAL(F) FROM U TO INF.                           L  21
C       (2) V**2<U<1, USES THE G-L QUADRATURE TO EVALUATE INTEGRAL(F)   L  22
C         FROM ONE TO INF AND A SERIES EXPANSION TO EVALUATE INTEGRAL(F)L  23
C         FROM U TO ONE.                                                L  24
C       (3) U<1, U<=V**2, USES THE REPRESENTATION INTEGRAL(F) FROM U    L  25
C         TO INF. = 2*K0(2*V)=INTEGRAL(F) FROM V**2/U TO INF.           L  26
C         EVALUATES THE ZERO ORDER MODIFIED BESSEL FUNCTION OF SECOND   L  27
C         KIND WITH IBM SUBROUTINE, EVALUATES INTEGRAL BY G-L QUAD.     L  28
C                                                                       L  29
C     ***************************************************************** L  30
      EXTERNAL FCT                                                      L  31
      REAL*8 U,V,Z,F,VV,SERIES                                          L  32
      COMMON /C1/ VV,Z                                                  L  33
      VV=V                                                              L  34
      IF (U-1.) 1,2,2                                                   L  35
C     CHECKS IF U<1                                                     L  36
    1 Z=V+V/U                                                           L  37
      IF (Z-1.) 3,4,4                                                   L  38
C     CHECKS IF V**2/U < 1                                              L  39
    2 Z=U                                                               L  40
      CALL DQL12(FCT,F)                                                 L  41
      L=F                                                               L  42
C     INTEGRAL U TO INF, EVALUATED BY GAUSS-LAGUERRE QUADRATURE         L  43
      GO TO 5                                                           L  44
    3 Z=1.                                                              L  45
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
      CALL DQL12(FCT,F)                                                  L  46
      L=F+SERIES(U,V)                                                    L  47
C     INTEGRAL 1 TO INF, BY G-L QUAD,, INTEGRAL U TO 1 BY SERIES EXP,    L  48
      GO TO 5                                                            L  49
    4 TWOV=2.*V                                                          L  50
      CALL BESK(TWOV,0,BK,IER)                                           L  51
      CALL DQL12(FCT,F)                                                  L  52
      L=2.*BK-F                                                          L  53
C     2K0(2V)=INTEGRAL V**2/U TO INF,                                    L  54
    5 RETURN                                                             L  55
      END                                                                L  56-

      REAL FUNCTION SERIES*8(U,V)                                        SER   1
C     ***************************************************************** SER   2
C                                                                        SER   3
C     FUNCTION SERIES                                                    SER   4
C                                                                        SER   5
C     PURPOSE                                                            SER   6
C        TO EVALUATE S(1)-S(U), WHERE S IS A SERIES EXPANSION OF         SER   7
C        INTEGRAL(EXP(-Y-V**2/Y)DY/Y) GIVEN BY; S= SUM, N=0 TO INFINITY, SER   8
C        (F(N)*SUM, N=0 TO INF,,(V**(2*N)/((N!)*(N+N)!))) WHERE F(N)=    SER   9
C        LOG(U) IF N=0 AND = ((-1)**N/N)*(U**N-(V**2/U)**N) IF N>0,      SER  10
C     DESCRIPTION OF PARAMETERS                                          SER  11
C        BOTH DOUBLE PRECISION                                           SER  12
C        U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE          SER  13
C            COEFFICIENT / 4*TRANSMISSIVITY * TIME                       SER  14
C        V = R/2*SQRT(K'/(T*B'))--ONE-HALF RADIAL DISTANCE*SQUARE ROOT   SER  15
C            (HYD. COND. OF CONFINING BED/TRANSMISSIVITY*THICKNESS       SER  16
C            OF CONFINING BED)                                           SER  17
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                      SER  18
C        NONE                                                            SER  19
C     METHOD                                                             SER  20
C        SUMMATION IS TERMINATED FOR THE INNER SERIES WHEN A TERM        SER  21
C        BECOMES LESS THAN 5.E-7/N AND FOR OUTER SERIES WHEN A TERM      SER  22
C        BECOMES LESS THAN 5.E-7                                         SER  23
C                                                                        SER  24
C     ***************************************************************** SER  25
      REAL*8 DLOG,DABS,S(2),VUM,UU                                       SER  26
      REAL*8 TEST,U,UM,EM,EN,SUM1,SUM,SIGN,V,VSQ,VSQU,RMUL,TERM,TERM1    SER  27
      TEST=5.D-07                                                        SER  28
      VSQ=V*V                                                            SER  29
      UU=U                                                               SER  30
      DO 6 I=1,2                                                         SER  31
C     EVALUATES SERIES FOR LOWER LIMIT = U AND UPPER LIMIT = 1           SER  32
      IF (I.EQ.2) U=1.                                                   SER  33
      UM=1.                                                              SER  34
      EM=-1.                                                             SER  35
      SUM1=0.                                                            SER  36
      SIGN=-1.                                                           SER  37
      VUM=1.                                                             SER  38
      VSQU=VSQ/U                                                         SER  39
    1 EM=EM+1.                                                           SER  40
      IF (EM-.1) 2,3,3                                                   SER  41
C     CHECKS FOR N=0                                                     SER  42
    2 RMUL=DLOG(U)                                                       SER  43
      TERM1=1.                                                           SER  44
      GO TO 4                                                            SER  45
    3 UM=UM*U                                                            SER  46
      IF (VUM.LT.1.D-30) VUM=0.                                          SER  47
      VUM=VUM*VSQU                                                       SER  48
      RMUL=(UM-VUM)/EM                                                   SER  49
      TERM1=TERM1/EM                                                     SER  50
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
    4 SIGN=-SIGN                                                          SER  51
      SUM=TERM1                                                           SER  52
      TERM=TERM1                                                          SER  53
      EN=0.                                                               SER  54
    5 EN=EN+1.                                                            SER  55
      TERM=TERM*VSQ/(EN*(EN+EM))                                          SER  56
      SUM=SUM+TERM                                                        SER  57
      IF (TEST.LE.DABS(RMUL*EN*TERM)) GO TO 5                             SER  58
C     TRUNCATES INNER SERIES IF OUTER TERM*N*INNER TERM < 5.E-7            SER  59
      SUM1=SUM1+SIGN*RMUL*SUM                                             SER  60
      IF (EM.LT..1) GO TO 1                                               SER  61
      IF (TEST.LE.DABS(RMUL*SUM)) GO TO 1                                 SER  62
C     TRUNCATES OUTER SERIES IF OUTER TERM*INNER SUM < 5.E-7               SER  63
    6 S(I)=SUM1                                                           SER  64
      U=UU                                                                SER  65
      SERIES=S(2)-S(1)                                                    SER  66
      RETURN                                                              SER  67
      END                                                                 SER  68-
      REAL FUNCTION FCT*8(X)                                              FCT   1
C     ********************************************************************FCT   2
C                                                                         FCT   3
C     FUNCTION FCT                                                        FCT   4
C                                                                         FCT   5
C     PURPOSE                                                             FCT   6
C        TO COMPUTE FCT(X)=EXP(-Z-V**2/(X+Z))/(X+Z)                       FCT   7
C     DESCRIPTION OF PARAMETERS                                           FCT   8
C        X - THE DOUBLE PRECISION VALUE OF X FOR WHICH FCT IS COMPUTED    FCT   9
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                       FCT  10
C        NONE                                                             FCT  11
C     METHOD                                                              FCT  12
C        FORTRAN EVALUATION OF FUNCTION                                   FCT  13
C                                                                         FCT  14
C     ********************************************************************FCT  15
      REAL*8 X,V,Z,P,DEXP                                                 FCT  16
      COMMON /C1/ V,Z                                                     FCT  17
      IF (X) 1,2,2                                                        FCT  18
    1 FCT=0.                                                              FCT  19
      GO TO 4                                                             FCT  20
    2 P=Z+V**2/(X+Z)                                                      FCT  21
      IF (P-5.D1) 3,3,1                                                   FCT  22
    3 FCT=DEXP(-P)/(X+Z)                                                  FCT  23
    4 RETURN                                                              FCT  24
      END                                                                 FCT  25-
      SUBROUTINE DGL12(FCT,Y)                                             DL12 380
C                                                                         DL12  10
C     ********************************************************************DL12  20
C                                                                         DL12  30
C        SUBROUTINE DGL12                                                 DL12  40
C                                                                         DL12  50
C        PURPOSE                                                          DL12  60
C           TO COMPUTE INTEGRAL(EXP(-X)*FCT(X), SUMMED OVER X             DL12  70
C                              FROM 0 TO INFINITY),                       DL12  80
C                                                                         DL12  90
C        USAGE                                                            DL12 100
C           CALL DGL12 (FCT,Y)                                            DL12 110
C           PARAMETER FCT REQUIRES AN EXTERNAL STATEMENT                  DL12 120
C                                                                         DL12 130
C        DESCRIPTION OF PARAMETERS                                        DL12 140
C           FCT    - THE NAME OF AN EXTERNAL DOUBLE PRECISION FUNCTION    DL12 150
C                    SUBPROGRAM USED.                                     DL12 160
C           Y      - THE RESULTING DOUBLE PRECISION INTEGRAL VALUE.       DL12 170
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
C                                                                       DL12 180
C     REMARKS                                                           DL12 190
C        NONE                                                           DL12 200
C                                                                       DL12 210
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                     DL12 220
C        THE EXTERNAL DOUBLE PRECISION FUNCTION SUBPROGRAM FCT(X)       DL12 230
C        MUST BE FURNISHED BY THE USER.                                 DL12 240
C                                                                       DL12 250
C     METHOD                                                            DL12 260
C        EVALUATION IS DONE BY MEANS OF 12-POINT GAUSSIAN-LAGUERRE      DL12 270
C        QUADRATURE FORMULA, WHICH INTEGRATES EXACTLY,                  DL12 280
C        WHENEVER FCT(X) IS A POLYNOMIAL UP TO DEGREE 23.               DL12 290
C        FOR REFERENCE, SEE                                             DL12 300
C        SHAO/CHEN/FRANK, TABLES OF ZEROS AND GAUSSIAN WEIGHTS OF       DL12 310
C        CERTAIN ASSOCIATED LAGUERRE POLYNOMIALS AND THE RELATED        DL12 320
C        GENERALIZED HERMITE POLYNOMIALS, IBM TECHNICAL REPORT          DL12 330
C        TR00.1100 (MARCH 1964), PP.24-25.                              DL12 340
C                                                                       DL12 350
C     ***************************************************************DL12 360
C                                                                       DL12 370
C                                                                       DL12 390
C                                                                       DL12 400
      DOUBLE PRECISION X,Y,FCT                                          DL12 410
C                                                                       DL12 420
      X=.3709912104446692 D2                                            DL12 430
      Y=.8148077467426240 D-15*FCT(X)                                   DL12 440
      X=.2848796725098400 D2                                            DL12 450
      Y=Y+.3061601635035021 D-11*FCT(X)                                 DL12 460
      X=.2215109037939701 D2                                            DL12 470
      Y=Y+.1342391030515004 D-8*FCT(X)                                  DL12 480
      X=.1711685518746226 D2                                            DL12 490
      Y=Y+.1668493876540910 D-6*FCT(X)                                  DL12 500
      X=.1300605499330635 D2                                            DL12 510
      Y=Y+.8365055856819800 D-5*FCT(X)                                  DL12 520
      X=.9621316842456867 D1                                            DL12 530
      Y=Y+.2032315926629994 D-3*FCT(X)                                  DL12 540
      X=.6844525453115177 D1                                            DL12 550
      Y=Y+.2663973541865316 D-2*FCT(X)                                  DL12 560
      X=.4599227639418348 D1                                            DL12 570
      Y=Y+.2010238115463410 D-1*FCT(X)                                  DL12 580
      X=.2833751337743507 D1                                            DL12 590
      Y=Y+.9044922221168090 D-1*FCT(X)                                  DL12 600
      X=.1512610269776419 D1                                            DL12 610
      Y=Y+.2440820113198776 D0*FCT(X)                                   DL12 620
      X=.6117574845151307 D0                                            DL12 630
      Y=Y+.3777592758731380 D0*FCT(X)                                   DL12 640
      X=.1157221173580207 D0                                            DL12 650
      Y=Y+.2647313710554432 D0*FCT(X)                                   DL12 660
      RETURN                                                            DL12 670
      END                                                               DL12 680-
      SUBROUTINE BESK(X,N,BK,IER)                                       BESK 410
C                                                                       BESK  10
C     ***************************************************************BESK  20
C                                                                       BESK  30
C        SUBROUTINE BESK                                                BESK  40
C                                                                       BESK  50
C           COMPUTE THE K BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDERBESK  60
C                                                                       BESK  70
C        USAGE                                                          BESK  80
C           CALL BESK(X,N,BK,IER)                                       BESK  90
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
C                                                                       BESK 100
C                                                                       BESK 110
C   DESCRIPTION OF PARAMETERS                                           BESK 120
C      X  =THE ARGUMENT OF THE K BESSEL FUNCTION DESIRED                BESK 130
C      N  =THE ORDER OF THE K BESSEL FUNCTION DESIRED                   BESK 140
C      BK =THE RESULTANT K BESSEL FUNCTION                              BESK 150
C      IER=RESULTANT ERROR CODE WHERE                                   BESK 160
C         IER=0  NO ERROR                                               BESK 170
C         IER=1  N IS NEGATIVE                                          BESK 180
C         IER=2  X IS ZERO OR NEGATIVE                                  BESK 190
C         IER=3  X .GT. 170, MACHINE RANGE EXCEEDED                     BESK 200
C         IER=4  BK .GT. 10**70                                         BESK 210
C                                                                       BESK 220
C   REMARKS                                                             BESK 230
C      N MUST BE GREATER THAN OR EQUAL TO ZERO                          BESK 240
C                                                                       BESK 250
C   SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                       BESK 260
C      NONE                                                             BESK 270
C                                                                       BESK 280
C   METHOD                                                              BESK 290
C      COMPUTES ZERO ORDER AND FIRST ORDER BESSEL FUNCTIONS USING       BESK 300
C      SERIES APPROXIMATIONS AND THEN COMPUTES N TH ORDER FUNCTION      BESK 310
C      USING RECURRENCE RELATION,                                       BESK 320
C      RECURRENCE RELATION AND POLYNOMIAL APPROXIMATION TECHNIQUE       BESK 330
C      AS DESCRIBED BY A,J,H,HITCHCOCK,'POLYNOMIAL APPROXIMATIONS       BESK 340
C      TO BESSEL FUNCTIONS OF ORDER ZERO AND ONE AND TO RELATED         BESK 350
C      FUNCTIONS', M,T,A,C., V,11,1957,PP,86-88, AND G,N, WATSON,       BESK 360
C      'A TREATISE ON THE THEORY OF BESSEL FUNCTIONS', CAMBRIDGE        BESK 370
C      UNIVERSITY PRESS, 1958, P, 62                                    BESK 380
C                                                                       BESK 390
C   ********************************************************************BESK 400
C                                                                       BESK 420
      DIMENSION T(12)                                                   BESK 430
      BK=.0                                                             BESK 440
      IF(N)10,11,11                                                     BESK 450
   10 IER=1                                                             BESK 460
      RETURN                                                            BESK 470
   11 IF(X)12,12,20                                                     BESK 480
   12 IER=2                                                             BESK 490
      RETURN                                                            BESK 500
   20 IF(X-170.0)22,22,21                                               BESK 510
   21 IER=3                                                             BESK 520
      RETURN                                                            BESK 530
   22 IER=0                                                             BESK 540
      IF(X-1.)36,36,25                                                  BESK 550
   25 A=EXP(-X)                                                         BESK 560
      B=1./X                                                            BESK 570
      C=SQRT(B)                                                         BESK 580
      T(1)=B                                                            BESK 590
      DO 26 L=2,12                                                      BESK 600
   26 T(L)=T(L-1)*B                                                     BESK 610
      IF(N-1)27,29,27                                                   BESK 620
C                                                                       BESK 630
C     COMPUTE KO USING POLYNOMIAL APPROXIMATION                         BESK 640
C                                                                       BESK 650
   27 GO=A*(1.2533141-.1566642*T(1)+.08811128*T(2)-.09139095*T(3)       BESK 660
     2+.1344596*T(4)-.2299850*T(5)+.3792410*T(6)-.5247277*T(7)          BESK 670
     3+.5575368*T(8)-.4262633*T(9)+.2184518*T(10)-.06680977*T(11)       BESK 680
     4+.009189383*T(12))*C                                              BESK 690
      IF(N)20,28,29                                                     BESK 700
   28 BK=GO                                                             BESK 710
      RETURN
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
C                                                                       BESK 720
C     COMPUTE K1 USING POLYNOMIAL APPROXIMATION                         BESK 730
C                                                                       BESK 740
   29 G1=A+(1.2533141+.4699927*T(1)-.1468583*T(2)+.1280427*T(3)         BESK 750
     2-.1736432*T(4)+.2847618*T(5)-.4594342*T(6)+.6283381*T(7)          BESK 760
     3-.6632295*T(8)+.5050239*T(9)-.2581304*T(10)+.07880001*T(11)       BESK 770
     4-.01062418*T(12))*C                                               BESK 780
      IF(N-1)20,30,31                                                   BESK 790
   30 BK=G1                                                             BESK 800
      RETURN                                                            BESK 810
C                                                                       BESK 820
C     FROM K0,K1 COMPUTE KN USING RECURRENCE RELATION                   BESK 830
C                                                                       BESK 840
   31 DO 35 J=2,N                                                       BESK 850
      GJ=2.*(FLOAT(J)-1.)*G1/X+G0                                       BESK 860
      IF(GJ-1.0E70)33,33,32                                             BESK 870
   32 IER=4                                                             BESK 880
      GO TO 34                                                          BESK 890
   33 G0=G1                                                             BESK 900
   35 G1=GJ                                                             BESK 910
   34 BK=GJ                                                             BESK 920
      RETURN                                                            BESK 930
   36 B=X/2.                                                            BESK 940
      A=.5772157+ALOG(B)                                                BESK 950
      C=B*B                                                             BESK 960
      IF(N-1)37,43,37                                                   BESK 970
C                                                                       BESK 980
C     COMPUTE K0 USING SERIES EXPANSION                                 BESK 990
C                                                                       BESK1000
   37 G0=-A                                                             BESK1010
      X2J=1.                                                            BESK1020
      FACT=1.                                                           BESK1030
      HJ=.0                                                             BESK1040
      DO 40 J=1,6                                                       BESK1050
      RJ=1./FLOAT(J)                                                    BESK1060
      IF(X2J.LT.1.E-40) X2J=0.                                          BESK1061
C     PREVIOUS STATEMENT ADDED TO IBM SUBROUTINE TO CORRECT UNDERFLOW   BESK1062
C     PROBLEM ON WATFOR COMPILER                                        BESK1063
      X2J=X2J*C                                                         BESK1070
      FACT=FACT*RJ*RJ                                                   BESK1080
      HJ=HJ+RJ                                                          BESK1090
   40 G0=G0+X2J*FACT*(HJ-A)                                             BESK1100
      IF(N)43,42,43                                                     BESK1110
   42 BK=G0                                                             BESK1120
      RETURN                                                            BESK1130
C                                                                       BESK1140
C     COMPUTE K1 USING SERIES EXPANSION                                 BESK1150
C                                                                       BESK1160
   43 X2J=B                                                             BESK1170
      FACT=1.                                                           BESK1180
      HJ=1.                                                             BESK1190
      G1=1./X+X2J*(.5+A-HJ)                                             BESK1200
      DO 50 J=2,8                                                       BESK1210
      X2J=X2J*C                                                         BESK1220
      RJ=1./FLOAT(J)                                                    BESK1230
      FACT=FACT*RJ*RJ                                                   BESK1240
      HJ=HJ+RJ                                                          BESK1250
   50 G1=G1+X2J*FACT*(.5+(A-HJ)*FLOAT(J))                               BESK1260
      IF(N-1)31,52,31                                                   BESK1270
   52 BK=G1                                                             BESK1280
      RETURN                                                            BESK1290
      END                                                               BESK1300
```

TABLE 2.1.—*Listing of program for partial penetration in a nonleaky artesian aquifer*—Continued

```
C                                                                       EXPI 350
      SUBROUTINE EXPI(X,RES,AUX)                                        EXPI  10
C     ******************************************************************EXPI  20
C     ******************************************************************EXPI  30
C                                                                       EXPI  40
C        SUBROUTINE EXPI                                                EXPI  50
C                                                                       EXPI  60
C        PURPOSE                                                        EXPI  70
C           COMPUTES THE EXPONENTIAL INTEGRAL -EI(-X)                   EXPI  80
C                                                                       EXPI  90
C        USAGE                                                          EXPI 100
C           CALL EXPI(X,RES)                                            EXPI 110
C                                                                       EXPI 120
C        DESCRIPTION OF PARAMETERS                                      EXPI 130
C           X      - ARGUMENT OF EXPONENTIAL INTEGRAL                   EXPI 140
C           RES    - RESULT VALUE                                       EXPI 150
C           AUX    - RESULTANT AUXILIARY VALUE                          EXPI 160
C                                                                       EXPI 170
C        REMARKS                                                        EXPI 180
C           X GT 170 (X LT -174) MAY CAUSE UNDERFLOW (OVERFLOW)         EXPI 190
C           WITH THE EXPONENTIAL FUNCTION                               EXPI 200
C           FOR X = 0 THE RESULT VALUE IS SET TO -1.E75                 EXPI 210
C                                                                       EXPI 220
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  EXPI 230
C           NONE                                                        EXPI 240
C                                                                       EXPI 250
C        METHOD                                                         EXPI 260
C           DEFINITION                                                  EXPI 270
C           RES=INTEGRAL(EXP(-T)/T, SUMMED OVER T FROM X TO INFINITY).  EXPI 280
C           EVALUATION                                                  EXPI 290
C           THREE DIFFERENT RATIONAL APPROXIMATIONS ARE USED IN THE     EXPI 300
C           RANGES 1 LE X, X LE -9 AND -9 LT X LE -3 RESPECTIVELY,      EXPI 310
C           A POLYNOMIAL APPROXIMATION IS USED IN -3 LT X LT 1.         EXPI 320
C                                                                       EXPI 330
C     ******************************************************************EXPI 340
C                                                                       EXPI 360
      IF(X-1.)2,1,1                                                     EXPI 370
    1 Y=1./X                                                            EXPI 380
      AUX=1.-Y*(((Y+3.377358E0)*Y+2.052156E0)*Y+2.709479E-1)/((((Y*     EXPI 390
     1 1.072553E0+5.716943E0)*Y+6.945239E0)*Y+2.593888E0)*Y+2.709496E-1) EXPI 400
      RES=AUX*Y*EXP(-X)                                                 EXPI 410
      RETURN                                                            EXPI 420
    2 IF(X+3.)6,6,3                                                     EXPI 430
    3 AUX=(((((((7.122452E-7*X-1.766345E-6)*X+2.928433E-5)*X-2.335379E-4EXPI 440
     1)*X+1.664158E-3)*X-1.041576E-2)*X+5.555682E-2)*X-2.500001E-1)*X   EXPI 450
     2+9.999999E-1                                                      EXPI 460
      RES=-1.E75                                                        EXPI 470
      IF(X)4,5,4                                                        EXPI 480
    4 RES=X*AUX-ALOG(ABS(X))-5.772157E-1                                EXPI 490
    5 RETURN                                                            EXPI 500
    6 IF(X+9.)8,8,7                                                     EXPI 510
    7 AUX=1.-((((5.176245E-2*X+3.061037E0)*X+3.243665E1)*X+2.244234E2)*XEXPI 520
     1+2.486697E2)/((((X+3.995161E0)*X+3.893944E1)*X+2.263818E1)*X      EXPI 530
     2+1.807837E2)                                                      EXPI 540
      GOTO 9                                                            EXPI 550
    8 Y=9./X                                                            EXPI 560
      AUX=1.-Y*(((Y+7.659824E-1)*Y-7.271015E-1)*Y-1.080693E0)/((((Y     EXPI 570
     1*2.518750E0+1.122927E1)*Y+5.921405E0)*Y-8.666702E0)*Y-9.724216E0) EXPI 580
    9 RES=AUX*EXP(-X)/X                                                 EXPI 590
      RETURN                                                            EXPI 60-
      END
```

TABLE 4.3—*Listing of program for radial flow in a leaky artesian aquifer*

```
C     ************************************************************************WUB   1
C                                                                               WUB   2
C     PURPOSE                                                                   WUB   3
C        TO COMPUTE A TABLE OF VALUES OF THE LEAKY AQUIFER WELL                 WUB   4
C        FUNCTION = W(U,R/B) = HANTUSH,M.S., AND JACOB,C.E., 1955,              WUB   5
C        NON-STEADY RADIAL FLOW IN AN INFINITE LEAKY AQUIFER: AM,               WUB   6
C        GEOPHYS. UNION TRANS., V. 36, NO. 1, P. 95-100.                        WUB   7
C     INPUT DATA                                                                WUB   8
C        1 CARD = FORMAT(2E10.5)                                                WUB   9
C             USMALL = SMALLEST VALUE OF 1/U FOR WHICH COMPUTATION IS           WUB  10
C             DESIRED,                                                          WUB  11
C             ULARGE = LARGEST  VALUE OF 1/U FOR WHICH COMPUTATION IS           WUB  12
C             DESIRED.                                                          WUB  13
C        2 CARDS = FORMAT(8E10.5)                                               WUB  14
C             BDAT = 12 VALUES OF R/B FOR TABLE.                                WUB  15
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                             WUB  16
C        L,SERIES,FCT,BESK,DQL12                                                WUB  17
C                                                                               WUB  18
C     ************************************************************************WUB  19
      REAL*4 L                                                                  WUB  20
      REAL*8 U,V                                                                WUB  21
      DIMENSION ARRAY(73,12), Y(73), BDAT(12), YNUM(6)                          WUB  22
      DATA YNUM/1.,1.5,2.,3.,5.,7./                                             WUB  23
      IRD=5                                                                     WUB  24
      IPT=6                                                                     WUB  25
      READ (IRD,6) USMALL,ULARGE                                                WUB  26
      READ (IRD,6) BDAT                                                         WUB  27
      IBEGIN=ALOG10(USMALL)                                                     WUB  28
      IEND=ALOG10(ULARGE)+.99999                                                WUB  29
      ILIMIT=(IEND-IBEGIN)*6+1                                                  WUB  30
      IF (ILIMIT.GT.73) ILIMIT=73                                               WUB  31
      DO 1 I=1,12                                                               WUB  32
      IF (BDAT(I).EQ.0.) GO TO 2                                                WUB  33
    1 CONTINUE                                                                  WUB  34
      NB=12                                                                     WUB  35
      GO TO 3                                                                   WUB  36
    2 NB=I-1                                                                    WUB  37
    3 II=0                                                                      WUB  38
      DO 4 I=1,ILIMIT                                                           WUB  39
      II=II+1                                                                   WUB  40
      IF (II.GT.6) II=1                                                         WUB  41
      IEXP=IBEGIN+(I-1)/6                                                       WUB  42
      Y(I)=YNUM(II)*10.**IEXP                                                   WUB  43
      U=1./Y(I)                                                                 WUB  44
      DO 4 J=1,NB                                                               WUB  45
      V=BDAT(J)/2.                                                              WUB  46
    4 ARRAY(I,J)=L(U,V)                                                         WUB  47
      WRITE (IPT,7) (BDAT(I),I=1,NB)                                            WUB  48
      DO 5 I=1,ILIMIT                                                           WUB  49
    5 WRITE (IPT,8) Y(I),(ARRAY(I,J),J=1,NB)                                    WUB  50
      STOP                                                                      WUB  51
C                                                                               WUB  52
C                                                                               WUB  53
    6 FORMAT (8E10.5)                                                           WUB  54
    7 FORMAT ('1','W(U,R/B)'/'0',10X,'I R/B'/' ',6X,'1/U I',12E10.2)            WUB  55
    8 FORMAT (' ',E10.3,12F10.4)                                                WUB  56
      END                                                                       WUB  57-
      REAL FUNCTION L*4(U,V)                                                     L   1
C     ************************************************************************  L   2
C                                                                               L   3
C     FUNCTION L                                                                L   4
C                                                                               L   5
```

TABLE 4.3—*Listing of program for radial flow in a leaky artesian aquifer*—Continued

```
C                                                                       L   6
C     PURPOSE                                                           L   7
C        TO COMPUTE THE INTEGRAL( EXP(-Y-V**2/Y)/Y) SUMMED OVER Y FROM  L   8
C        U TO INFINITY(WELL FUNCTION FOR LEAKY AQUIFERS).               L   9
C     DESCRIPTION OF PARAMETERS                                         L  10
C        BOTH DOUBLE PRECISION                                          L  11
C        U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE         L  12
C            COEFFICIENT / 4*TRANSMISSIVITY * TIME                      L  13
C        V = R/2*SQRT(K'/(T*B'))--ONE-HALF RADIAL DISTANCE*SQUARE ROOT  L  14
C            (HYD. COND. OF CONFINING BED/TRANSMISSIVITY*THICKNESS      L  15
C            OF CONFINING BED)                                          L  16
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                     L  17
C        DQL12,SERIES,BESK,FCT                                          L  18
C     METHOD                                                            L  19
C        IN THE FOLLOWING F=EXP(-Y-V**2/Y)/Y                            L  20
C     (1) U>=1, USES A GAUSSIAN-LAGUERRE QUADRATURE FORMULA TO          L  21
C       EVALUATE INTEGRAL(F) FROM U TO INF.                             L  22
C     (2) V**2<U<1, USES THE G-L QUADRATURE TO EVALUATE INTEGRAL(F)     L  23
C       FROM ONE TO INF AND A SERIES EXPANSION TO EVALUATE INTEGRAL(F)  L  24
C       FROM U TO ONE.                                                  L  25
C     (3) U<1, U<=V**2, USES THE REPRESENTATION INTEGRAL(F) FROM U      L  26
C       TO INF. = 2*K0(2*V)-INTEGRAL(F) FROM V**2/U TO INF.             L  27
C       EVALUATES THE ZERO ORDER MODIFIED BESSEL FUNCTION OF SECOND     L  28
C       KIND WITH IBM SUBROUTINE, EVALUATES INTEGRAL BY G-L QUAD.       L  29
C                                                                       L  30
C     ********************************************************************L  31
      EXTERNAL FCT                                                      L  32
      REAL*8 U,V,Z,F,VV,SERIES                                          L  33
      COMMON /C1/ VV,Z                                                  L  34
      VV=V                                                              L  35
      IF (U-1.) 1,2,2                                                   L  36
C     CHECKS IF U<1                                                     L  37
    1 Z=V*V/U                                                           L  38
      IF (Z-1.) 3,4,4                                                   L  39
C     CHECKS IF V**2/U < 1                                              L  40
    2 Z=U                                                               L  41
      CALL DQL12(FCT,F)                                                 L  42
      L=F                                                               L  43
C     INTEGRAL U TO INF, EVALUATED BY GAUSS-LAGUERRE QUADRATURE         L  44
      GO TO 5                                                           L  45
    3 Z=1.                                                              L  46
      CALL DQL12(FCT,F)                                                 L  47
      L=F+SERIES(U,V)                                                   L  48
C     INTEGRAL 1 TO INF, BY G-L QUAD., INTEGRAL U TO 1 BY SERIES EXP.   L  49
      GO TO 5                                                           L  50
    4 TWOV=2.*V                                                         L  51
      CALL BESK(TWOV,0,BK,IER)                                          L  52
      CALL DQL12(FCT,F)                                                 L  53
      L=2.*BK-F                                                         L  54
C     2K0(2V)-INTEGRAL V**2/U TO INF.                                   L  55
    5 RETURN                                                            L  56-
      END                                                             SER   1
      REAL FUNCTION SERIES*8(U,V)                                     SER   2
C     ********************************************************************SER   3
C                                                                       SER   4
C     FUNCTION SERIES                                                   SER   5
C                                                                       SER   6
C     PURPOSE                                                           SER   7
C        TO EVALUATE S(1)=S(U), WHERE S IS A SERIES EXPANSION OF        SER   8
C        INTEGRAL(EXP(-Y-V**2/Y)DY/Y) GIVEN BY: S= SUM, M=0 TO INFINITY,SER   9
C        (F(M)*SUM, N=0 TO INF.,(V**(2*N)/((N!)*(M+N)!)) WHERE F(M)=    SER  10
C        LOG(U) IF M=0 AND = ((-1)**M/M!)*(U**M-(V**2/U)**M) IF M>0.    SER  11
C     DESCRIPTION OF PARAMETERS                                         SER  12
C        BOTH DOUBLE PRECISION                                          SER  13
C        U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE         SER  14
C            COEFFICIENT / 4*TRANSMISSIVITY * TIME                      SER  15
C        V = R/2*SQRT(K'/(T*B'))--ONE-HALF RADIAL DISTANCE*SQUARE ROOT  SER  16
C            (HYD. COND. OF CONFINING BED/TRANSMISSIVITY*THICKNESS      SER  17
C            OF CONFINING BED)
```

TABLE 4.3—*Listing of program for radial flow in a leaky artesian aquifer*—Continued

```
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                          SER  18
C        NONE                                                                SER  19
C     METHOD                                                                 SER  20
C        SUMMATION IS TERMINATED FOR THE INNER SERIES WHEN A TERM            SER  21
C        BECOMES LESS THAN 5.E-7/N AND FOR OUTER SERIES WHEN A TERM          SER  22
C        BECOMES LESS THAN 5.E-7                                             SER  23
C                                                                            SER  24
C     ************************************************************************SER  25
      REAL*8 DLOG,DABS,S(2),VUM,UU                                           SER  26
      REAL*8 TEST,U,UM,EM,EN,SUM1,SUM,SIGN,V,VSQ,VSQU,RMUL,TERM,TERM1        SER  27
      TEST=5.D-07                                                            SER  28
      VSQ=V*V                                                                SER  29
      UU=U                                                                   SER  30
      DO 6 I=1,2                                                             SER  31
C     EVALUATES SERIES FOR LOWER LIMIT = U AND UPPER LIMIT = 1               SER  32
      IF (I.EQ.2) U=1.                                                       SER  33
      UM=1.                                                                  SER  34
      EM=-1.                                                                 SER  35
      SUM1=0.                                                                SER  36
      SIGN=-1.                                                               SER  37
      VUM=1.                                                                 SER  38
      VSQU=VSQ/U                                                             SER  39
    1 EM=EM+1.                                                               SER  40
      IF (EM.) 2,3,3                                                         SER  41
C     CHECKS FOR M=0                                                         SER  42
    2 RMUL=DLOG(U)                                                           SER  43
      TERM1=1.                                                               SER  44
      GO TO 4                                                                SER  45
    3 UM=UM*U                                                                SER  46
      IF (VUM.LT.1.D-30) VUM=0.                                              SER  47
      VUM=VUM*VSQU                                                           SER  48
      RMUL=(UM*VUM)/EM                                                       SER  49
      TERM1=TERM1/EM                                                         SER  50
    4 SIGN=-SIGN                                                             SER  51
      SUM=TERM1                                                              SER  52
      TERM=TERM1                                                             SER  53
      EN=0.                                                                  SER  54
    5 EN=EN+1.                                                               SER  55
      TERM=TERM*VSQ/(EN*(EN+EM))                                             SER  56
      SUM=SUM+TERM                                                           SER  57
      IF (TEST.LE.DABS(RMUL*EN*TERM)) GO TO 5                                SER  58
C     TRUNCATES INNER SERIES IF OUTER TERM*N*INNER TERM < 5.E-7               SER  59
      SUM1=SUM1+SIGN*RMUL*SUM                                                SER  60
      IF (EM.LT.1.) GO TO 1                                                  SER  61
      IF (TEST.LE.DABS(RMUL*SUM)) GO TO 1                                    SER  62
C     TRUNCATES OUTER SERIES IF OUTER TERM*INNER SUM < 5.E-7                  SER  63
    6 S(I)=SUM1                                                              SER  64
      U=UU                                                                   SER  65
      SERIES=S(2)-S(1)                                                       SER  66
      RETURN                                                                 SER  67
      END                                                                    SER  68.
      REAL FUNCTION FCT*8(X)                                                 FCT   1
C     ************************************************************************FCT   2
C                                                                            FCT   3
C     FUNCTION FCT                                                           FCT   4
C                                                                            FCT   5
C     PURPOSE                                                                FCT   6
C        TO COMPUTE FCT(X)=EXP(-Z-V**2/(X+Z))/(X+Z)                          FCT   7
```

TABLE 4.3—*Listing of program for radial flow in a leaky artesian aquifer*—Continued

```
C                                                                       FCT   8
C     DESCRIPTION OF PARAMETERS                                         FCT   9
C        X - THE DOUBLE PRECISION VALUE OF X FOR WHICH FCT IS COMPUTED  FCT  10
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                     FCT  11
C        NONE                                                           FCT  12
C     METHOD                                                            FCT  13
C        FORTRAN EVALUATION OF FUNCTION                                 FCT  14
C     ***************************************************************FCT  15
      REAL*8 X,V,Z,P,DEXP                                               FCT  16
      COMMON /C1/ V,Z                                                   FCT  17
      IF (X) 1,2,2                                                      FCT  18
    1 FCT=0.                                                            FCT  19
      GO TO 4                                                           FCT  20
    2 P=Z+V**2/(X+Z)                                                    FCT  21
      IF (P-5.01) 3,3,1                                                 FCT  22
    3 FCT=DEXP(-P)/(X+Z)                                                FCT  23
    4 RETURN                                                            FCT  24
      END                                                               FCT  25-
      SUBROUTINE DQL12(FCT,Y)                                           DL12 380
C     ***************************************************************DL12  10
C                                                                       DL12  20
C        SUBROUTINE DQL12                                               DL12  30
C                                                                       DL12  40
C        PURPOSE                                                        DL12  50
C           TO COMPUTE INTEGRAL(EXP(-X)*FCT(X), SUMMED OVER X           DL12  60
C                              FROM 0 TO INFINITY),                     DL12  70
C                                                                       DL12  80
C        USAGE                                                          DL12  90
C           CALL DQL12 (FCT,Y)                                          DL12 100
C           PARAMETER FCT REQUIRES AN EXTERNAL STATEMENT                DL12 110
C                                                                       DL12 120
C        DESCRIPTION OF PARAMETERS                                      DL12 130
C           FCT    - THE NAME OF AN EXTERNAL DOUBLE PRECISION FUNCTION  DL12 140
C                    SUBPROGRAM USED.                                   DL12 150
C           Y      - THE RESULTING DOUBLE PRECISION INTEGRAL VALUE.     DL12 160
C                                                                       DL12 170
C        REMARKS                                                        DL12 180
C           NONE                                                        DL12 190
C                                                                       DL12 200
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  DL12 210
C           THE EXTERNAL DOUBLE PRECISION FUNCTION SUBPROGRAM FCT(X)    DL12 220
C           MUST BE FURNISHED BY THE USER.                              DL12 230
C                                                                       DL12 240
C        METHOD                                                         DL12 250
C           EVALUATION IS DONE BY MEANS OF 12-POINT GAUSSIAN-LAGUERRE   DL12 260
C           QUADRATURE FORMULA, WHICH INTEGRATES EXACTLY,               DL12 270
C           WHENEVER FCT(X) IS A POLYNOMIAL UP TO DEGREE 23.            DL12 280
C           FOR REFERENCE, SEE                                          DL12 290
C           SHAO/CHEN/FRANK, TABLES OF ZEROS AND GAUSSIAN WEIGHTS OF    DL12 300
C           CERTAIN ASSOCIATED LAGUERRE POLYNOMIALS AND THE RELATED     DL12 310
C           GENERALIZED HERMITE POLYNOMIALS, IBM TECHNICAL REPORT       DL12 320
C           TR00.1100 (MARCH 1964), PP.24-25.                           DL12 330
C                                                                       DL12 340
C     ***************************************************************DL12 350
C                                                                       DL12 360
      DOUBLE PRECISION X,Y,FCT                                          DL12 370
C                                                                       DL12 390
      X=.3709912104446692 D2                                            DL12 400
      Y=.8148077467426240 D-15*FCT(X)                                   DL12 410
                                                                        DL12 420
                                                                        DL12 430
                                                                        DL12 440
```

TABLE 4.3—*Listing of program for radial flow in a leaky artesian aquifer*—Continued

```
      X=.28487967250984O0 D2                                                DL12 450
      Y=Y+.3061601635035021 D-11*FCT(X)                                     DL12 460
      X=.2215109037939701 D2                                                DL12 470
      Y=Y+.1342391030515004 D-8*FCT(X)                                      DL12 480
      X=.1711685518746226 D2                                                DL12 490
      Y=Y+.1668493876540910 D-6*FCT(X)                                      DL12 500
      X=.1300605499330635 D2                                                DL12 510
      Y=Y+.8365055856819B0 D-5*FCT(X)                                       DL12 520
      X=.962131684245687 D1                                                 DL12 530
      Y=Y+.2032315926629994 D-3*FCT(X)                                      DL12 540
      X=.6844525453115177 D1                                                DL12 550
      Y=Y+.2663973541865316 D-2*FCT(X)                                      DL12 560
      X=.4599227639418348 D1                                                DL12 570
      Y=Y+.2010238115463410 D-1*FCT(X)                                      DL12 580
      X=.2833751337743507 D1                                                DL12 590
      Y=Y+.9044922221166809 D-1*FCT(X)                                      DL12 600
      X=.1512610269776419 D1                                                DL12 610
      Y=Y+.2440820113198776 D0*FCT(X)                                       DL12 620
      X=.6117574845151307 D0                                                DL12 630
      Y=Y+.3777592758731380 D0*FCT(X)                                       DL12 640
      X=.1157221173580207 D0                                                DL12 650
      Y=Y+.2647313710554432 D0*FCT(X)                                       DL12 660
      RETURN                                                                DL12 670
      END                                                                   DL12 680
      SUBROUTINE BESK(X,N,BK,IER)                                           BESK 410
C                                                                           BESK  10
C.....BESK  20
C     SUBROUTINE BESK                                                       BESK  30
C                                                                           BESK  40
C        COMPUTE THE K BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDER       BESK  50
C                                                                           BESK  60
C     USAGE                                                                 BESK  70
C        CALL BESK(X,N,BK,IER)                                              BESK  80
C                                                                           BESK  90
C     DESCRIPTION OF PARAMETERS                                             BESK 100
C        X  =THE ARGUMENT OF THE K BESSEL FUNCTION DESIRED                  BESK 110
C        N  =THE ORDER OF THE K BESSEL FUNCTION DESIRED                     BESK 120
C        BK =THE RESULTANT K BESSEL FUNCTION                                BESK 130
C        IER=RESULTANT ERROR CODE WHERE                                     BESK 140
C           IER=0  NO ERROR                                                 BESK 150
C           IER=1  N IS NEGATIVE                                            BESK 160
C           IER=2  X IS ZERO OR NEGATIVE                                    BESK 170
C           IER=3  X .GT. 170, MACHINE RANGE EXCEEDED                       BESK 180
C           IER=4  BK .GT. 10**70                                           BESK 190
C                                                                           BESK 200
C     REMARKS                                                               BESK 210
C        N MUST BE GREATER THAN OR EQUAL TO ZERO                            BESK 220
C                                                                           BESK 230
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                         BESK 240
C        NONE                                                               BESK 250
C                                                                           BESK 260
C     METHOD                                                                BESK 270
C                                                                           BESK 280
C        COMPUTES ZERO ORDER AND FIRST ORDER BESSEL FUNCTIONS USING         BESK 290
C        SERIES APPROXIMATIONS AND THEN COMPUTES N TH ORDER FUNCTION        BESK 300
C        USING RECURRENCE RELATION,                                         BESK 310
C        RECURRENCE RELATION AND POLYNOMIAL APPROXIMATION TECHNIQUE         BESK 320
C        AS DESCRIBED BY A.J.M.HITCHCOCK,'POLYNOMIAL APPROXIMATIONS         BESK 330
C        TO BESSEL FUNCTIONS OF ORDER ZERO AND ONE AND TO RELATED           BESK 340
C        FUNCTIONS', M.T.A.C., V.11,1957,PP.86-88, AND G.N. WATSON,         BESK 350
C        'A TREATISE ON THE THEORY OF BESSEL FUNCTIONS', CAMBRIDGE          BESK 360
C        UNIVERSITY PRESS, 1958, P. 62                                      BESK 370
```

TABLE 4.3—*Listing of program for radial flow in a leaky artesian aquifer*—Continued

```
                                                                      BESK 380
C                                                                     BESK 390
C     ****************************************************************BESK 400
C                                                                     BESK 420
      DIMENSION T(12)                                                 BESK 430
      BK=.0                                                           BESK 440
      IF(N)10,11,11                                                   BESK 450
   10 IER=1                                                           BESK 460
      RETURN                                                          BESK 470
   11 IF(X)12,12,20                                                   BESK 480
   12 IER=2                                                           BESK 490
      RETURN                                                          BESK 500
   20 IF(X-170.0)22,22,21                                             BESK 510
   21 IER=3                                                           BESK 520
      RETURN                                                          BESK 530
   22 IER=0                                                           BESK 540
      IF(X-1.)36,36,25                                                BESK 550
   25 A=EXP(-X)                                                       BESK 560
      B=1./X                                                          BESK 570
      C=SQRT(B)                                                       BESK 580
      T(1)=B                                                          BESK 590
      DO 26 L=2,12                                                    BESK 600
   26 T(L)=T(L-1)*B                                                   BESK 610
      IF(N-1)27,29,27                                                 BESK 620
C                                                                     BESK 630
C     COMPUTE K0 USING POLYNOMIAL APPROXIMATION                       BESK 640
C                                                                     BESK 650
   27 G0=A*(1.2533141-.1566642*T(1)+.08811128*T(2)-.09139095*T(3)     BESK 660
     2+.1344596*T(4)-.2299850*T(5)+.3792410*T(6)-.5247277*T(7)        BESK 670
     3+.5575368*T(8)-.4262633*T(9)+.2184518*T(10)-.06680977*T(11)     BESK 680
     4+.009189383*T(12))*C                                            BESK 690
      IF(N)20,28,29                                                   BESK 700
   28 BK=G0                                                           BESK 710
      RETURN                                                          BESK 720
C                                                                     BESK 730
C     COMPUTE K1 USING POLYNOMIAL APPROXIMATION                       BESK 740
C                                                                     BESK 750
   29 G1=A*(1.2533141+.4699927*T(1)-.1468583*T(2)+.1280427*T(3)       BESK 760
     2-.1736432*T(4)+.2847618*T(5)-.4594342*T(6)+.6283381*T(7)        BESK 770
     3-.6632295*T(8)+.5050239*T(9)-.2581304*T(10)+.07880001*T(11)     BESK 780
     4-.01062416*T(12))*C                                             BESK 790
      IF(N-1)20,30,31                                                 BESK 800
   30 BK=G1                                                           BESK 810
      RETURN                                                          BESK 820
C                                                                     BESK 830
C     FROM K0,K1 COMPUTE KN USING RECURRENCE RELATION                 BESK 840
C                                                                     BESK 850
   31 DO 35 J=2,N                                                     BESK 860
      GJ=2.*(FLOAT(J)-1.)*G1/X+G0                                     BESK 870
      IF(GJ-1.0E70)33,33,32                                           BESK 880
   32 IER=4                                                           BESK 890
      GO TO 34                                                        BESK 900
   33 G0=G1                                                           BESK 910
   35 G1=GJ                                                           BESK 920
   34 BK=GJ                                                           BESK 930
      RETURN                                                          BESK 940
   36 B=X/2.                                                          BESK 950
      A=.5772157+ALOG(B)                                              BESK 960
      C=B*B                                                           BESK 970
      IF(N-1)37,43,37                                                 BESK 980
C                                                                     BESK 990
C     COMPUTE K0 USING SERIES EXPANSION
```

TABLE 4.3—*Listing of program for radial flow in a leaky artesian aquifer*—Continued

```
C
   37 G0=-A                                                              BESK1000
      X2J=1.                                                             BESK1010
      FACT=1.                                                            BESK1020
      HJ=.0                                                              BESK1030
      DO 40 J=1,6                                                        BESK1040
      RJ=1./FLOAT(J)                                                     BESK1050
      IF(X2J.LT.1.E-40) X2J=0.                                           BESK1060
C     PREVIOUS STATEMENT ADDED TO IBM SUBROUTINE TO CORRECT UNDERFLOW    BESK1061
C     PROBLEM ON WATFOR COMPILER                                         BESK1062
      X2J=X2J*C                                                          BESK1063
      FACT=FACT*RJ*RJ                                                    BESK1070
      HJ=HJ+RJ                                                           BESK1080
   40 G0=G0+X2J*FACT*(HJ-A)                                              BESK1090
      IF(N)43,42,43                                                      BESK1100
   42 BK=G0                                                              BESK1110
      RETURN                                                             BESK1120
C                                                                        BESK1130
C     COMPUTE K1 USING SERIES EXPANSION                                  BESK1140
C                                                                        BESK1150
   43 X2J=B                                                              BESK1160
      FACT=1.                                                            BESK1170
      HJ=1.                                                              BESK1180
      G1=1./X+X2J*(.5+A-HJ)                                              BESK1190
      DO 50 J=2,8                                                        BESK1200
      X2J=X2J*C                                                          BESK1210
      RJ=1./FLOAT(J)                                                     BESK1220
      FACT=FACT*RJ*RJ                                                    BESK1230
      HJ=HJ+RJ                                                           BESK1240
   50 G1=G1+X2J*FACT*(.5+(A-HJ)*FLOAT(J))                                BESK1250
      IF(N-1)51,52,51                                                    BESK1260
   52 BK=G1                                                              BESK1270
      RETURN                                                             BESK1280
      END                                                                BESK1290
                                                                         BESK130-
```

TABLE 5.2—*Listing of program for radial flow in a leaky artesian aquifer with storage of water in the confining beds*

```
C     ***************************************************************LST   1
C                                                                        LST   2
C     PURPOSE                                                            LST   3
C        TO COMPUTE TYPE CURVE FUNCTION VALUES FOR W(U,BETA) --          LST   4
C        HANTUSH,M.S.,1960, MODIFICATION OF THE THEORY OF LEAKY          LST   5
C        AQUIFERS: JOUR. GEOPHYS. RES., V. 65, NO. 11, P. 3713-3725.     LST   6
C        THE COMPUTATIONAL ALGORITHM WAS DEVISED AND PROGRAMMED BY       LST   7
C        S.S.PAPADOPULUS.                                                LST   8
C     INPUT DATA                                                         LST   9
C        1 CARD - FORMAT(2E10.5)                                         LST  10
C             USMALL = SMALLEST(BEGINNING) VALUE OF 1/U.                 LST  11
C             ULARGE = LARGEST(ENDING) VALUE OF 1/U.                     LST  12
C        2 CARDS - FORMAT(8E10.5)                                        LST  13
C             BDAT = 12 VALUES OF BETA (ZERO OR BLANK VALUES ARE         LST  14
C             PERMISSIBLE IF LESS THAN 12 DESIRED, WILL TERMINATE        LST  15
C             AT FIRST ZERO OR BLANK VALUE).                             LST  16
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                      LST  17
C        W,DQG32,WUR,W - MUST BE INCLUDED IN DECK.                       LST  18
C        DSQRT,DEXP,DERFC,DLOG - MUST BE IN COMPUTER LIBRARY.            LST  19
C                                                                        LST  20
C     ***************************************************************LST  21
      REAL*8 U,BETA,W                                                    LST  22
      DIMENSION ARRAY(73,12), Y(73), BDAT(12), YNUM(6)                   LST  23
      DATA YNUM/1.,1.5,2.,3.,5.,7./                                      LST  24
      IRD=5                                                              LST  25
      IPT=6                                                              LST  26
      READ (IRD,6) USMALL,ULARGE                                         LST  27
      READ (IRD,6) BDAT                                                  LST  28
      IBEGIN=ALOG10(USMALL)                                              LST  29
      IEND=ALOG10(ULARGE)+.99999                                         LST  30
      ILIMIT=(IEND-IBEGIN)*8+1                                           LST  31
      IF (ILIMIT.GT.73) ILIMIT=73                                        LST  32
      DO 1 I=1,12                                                        LST  33
      IF (BDAT(I).EQ.0.) GO TO 2                                         LST  34
    1 CONTINUE                                                           LST  35
      NB=12                                                              LST  36
```

TABLE 5.2—*Listing of program for radial flow in a leaky artesian aquifer with storage of water in the confining beds*—Continued

```
      GO TO 3                                                          LST  37
    2 NB=I-1                                                           LST  38
      II=0                                                             LST  39
    3 DO 4 I=1,ILIM1                                                   LST  40
      IEXP=IBEGIN+(I-1)/6                                              LST  41
      II=II+1                                                          LST  42
      IF (II.GT.6) II=1                                                LST  43
      Y(I)=YNUM(II)*10.**IEXP                                          LST  44
      U=1./Y(I)                                                        LST  45
      DO 4 J=1,NB                                                      LST  46
      BETA=BDAT(J)                                                     LST  47
    4 ARRAY(I,J)=HM(U,BETA)                                            LST  48
      WRITE (IPT,7) (BDAT(I),I=1,NB)                                   LST  49
      DO 5 I=1,ILIM1                                                   LST  50
    5 WRITE (IPT,8) Y(I),(ARRAY(I,J),J=1,NB)                           LST  51
      STOP                                                             LST  52
C                                                                      LST  53
    6 FORMAT (8E10.5)                                                  LST  54
    7 FORMAT ('1','HM(U,BETA)'/'0',10X,'I BETA'/' ',6X,'1/U ',12E10.2) LST  55
    8 FORMAT (' ',E10.3,12F10.4)                                       LST  56
      END                                                              LST  57*
      DOUBLE PRECISION FUNCTION H(U,B)                                   H    1
C     ***************************************************************  H    2
C                                                                        H    3
C     FUNCTION H                                                         H    4
C     PURPOSE                                                            H    5
C        TO COMPUTE THE INTEGRAL OF                                      H    6
C        EXP(-Y)*ERFC(B*SQRT(U)/SQRT(Y*(Y-U)))/Y SUMMED OVER Y           H    7
C        FROM U TO INFINITY (FUNCTION H(U,BETA) OF HANTUSH).             H    8
C     DESCRIPTION OF PARAMETERS                                          H    9
C        BOTH DOUBLE PRECISION                                           H   10
C        U = R**2*S/(4*T*TIME), (RADIAL DISTANCE SQUARED * STORAGE       H   11
C        COEFFICIENT / (4 * TRANSMISSIVITY * TIME), U MUST BE > 1.D-60,  H   12
C        B = (R/4)*(SQRT(K'*S'/(S'*T*S)+K''*S''/(S''*T*S)),              H   13
C              K',S',S' = HYD. COND., STORAGE COEFF., THICKNESS OF       H   14
C                   UPPER CONFINING BED,                                 H   15
C              K'',S'',S'' = HYD. COND., STORAGE COEFF., THICKNESS OF    H   16
C                   LOWER CONFINING BED.                                 H   17
C                                                                        H   18
C     METHOD                                                             H   19
C        I. FOR U < 1.D-60, NO COMPUTATION IS MADE,                      H   20
C        II. FOR B=0, H(U,0)=W(U) (THEIS WELL FUNCTION),                 H   21
C        III. H(U,B)=0 IF                                                H   22
C             1. U > 10,                                                 H   23
C             2. B > 1 AND B**2*U > 300,                                 H   24
C        IV. ERFC(ARG)=0 FOR ARG > 40 AND H(U,B) = H(UB,B)               H   25
C             FOR U < Y < UB WHERE UB IS THE U CORRESPONDING TO ARG = 40 H   26
C             SINCE H(UB,B) < W(UB) THEN FOR UB > 10, H(U,B) = 0,        H   27
C             ERFC(ARG) = 1 FOR ARG < 2.E-10 AND H(UUB,B) = W(UUB)       H   28
C             WHERE UUB IS THE U CORRESPONDING TO ARG = 2.E-10,          H   29
C             IF UUB > 10, H(U,B) = INTEGRAL FROM UB TO 10,              H   30
C             IF UUB < 10, H(U,B) = INTEGRAL FROM UB TO UUB + W(UUB)     H   31
C                                                                        H   32
C     ***************************************************************  H   33
      IMPLICIT REAL*8(A-H,O-Z)                                           H   34
      COMMON UUU,BBB                                                     H   35
      EXTERNAL HUB                                                       H   36
      UUU=U                                                              H   37
      BBB=B                                                              H   38
      IF (U.GT.1.D-60) GO TO 1                                           H   39
      WRITE (6,7)                                                        H   40
      STOP                                                               H   41
    1 IF (B.EQ.0.0) GO TO 5                                              H   42
      IF (U.ST.10.0) GO TO 6                                             H   43
      BU=B*B*U                                                           H   44
      IF (B.GT.1.0.AND.BU.GE.300.0) GO TO 6                              H   45
      H1=0.0                                                             H   46
      UP=10.0                                                            H   47
      UB=0.5*U*(1.0+DSQRT(1.0+0.025*B*B/U))                              H   48
      IF (UB.GT.UP) GO TO 6                                              H   49
      UUB=0.5*U*(1.0+DSQRT(1.0+1.D20*B*B/U))                             H   50
      IF (UUB.GT.UP) GO TO 2                                             H   51
      H1=W(UUB)                                                          H   52
      UP=UUB                                                             H   53
    2 H2=0.0                                                             H   54
      XL=UB                                                              H   55
    3 XU=10.*XL                                                          H   56
      IF (XU.GE.UP) XU=UP                                                H   57
      CALL DUG32(XL,XU,HUB,AREA)                                         H   58
      H2=H2+AREA                                                         H   59
      XL=XU                                                              H   60
      IF (XL.EQ.UP) GO TO 4                                              H   61
      GO TO 3                                                            H   62
    4 H=H1+H2                                                            H   63
      RETURN                                                             H   64
    5 H=W(U)                                                             H   65
      RETURN
```

TABLE 5.2—*Listing of program for radial flow in a leaky artesian aquifer with storage of water in the confining beds*—Continued

```
      6 HB=0.0                                                                M   66
        RETURN                                                                M   67
C                                                                             M   68
      7 FORMAT ('0','U TOO SMALL FOR COMPUTATION')                            M   69
        END                                                                   M   70
        SUBROUTINE DQG32(XL,XU,FCT,Y)                                         DQG   1
C                                                                             DQG   2
C     ***************************************************************       DQG   3
C                                                                             DQG   4
C        SUBROUTINE DQG32                                                     DQG   5
C                                                                             DQG   6
C        PURPOSE                                                              DQG   7
C           TO COMPUTE INTEGRAL(FCT(X), SUMMED OVER X FROM XL TO XU)          DQG   8
C                                                                             DQG   9
C        USAGE                                                                DQG  10
C           CALL DQG32 (XL,XU,FCT,Y)                                          DQG  11
C           PARAMETER FCT REQUIRES AN EXTERNAL STATEMENT                      DQG  12
C                                                                             DQG  13
C        DESCRIPTION OF PARAMETERS                                            DQG  14
C           XL     - DOUBLE PRECISION LOWER BOUND OF THE INTERVAL,            DQG  15
C           XU     - DOUBLE PRECISION UPPER BOUND OF THE INTERVAL,            DQG  16
C           FCT    - THE NAME OF AN EXTERNAL DOUBLE PRECISION FUNCTION        DQG  17
C                    SUBPROGRAM USED,                                         DQG  18
C           Y      - THE RESULTING DOUBLE PRECISION INTEGRAL VALUE,           DQG  19
C                                                                             DQG  20
C        REMARKS                                                              DQG  21
C           NONE                                                              DQG  22
C                                                                             DQG  23
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                        DQG  24
C           THE EXTERNAL DOUBLE PRECISION FUNCTION SUBPROGRAM FCT(X)          DQG  25
C           MUST BE FURNISHED BY THE USER,                                    DQG  26
C                                                                             DQG  27
C        METHOD                                                               DQG  28
C           EVALUATION IS DONE BY MEANS OF 32-POINT GAUSS QUADRATURE          DQG  29
C           FORMULA, WHICH INTEGRATES POLYNOMIALS UP TO DEGREE 63             DQG  30
C           EXACTLY, FOR REFERENCE, SEE                                       DQG  31
C           V.I.KRYLOV, APPROXIMATE CALCULATION OF INTEGRALS,                 DQG  32
C           MACMILLAN, NEW YORK/LONDON, 1962, PP.100-111 AND 337-340,         DQG  33
C                                                                             DQG  34
C     ***************************************************************       DQG  35
C                                                                             DQG  36
      DOUBLE PRECISION XL,XU,Y,A,B,C,FCT                                      DQG  37
      A=.500*(XU+XL)                                                          DQG  38
      B=XU-XL                                                                 DQG  39
      C=.49863193092474080D+0*B                                               DQG  40
      Y=.35093050047350480+2*(FCT(A+C)+FCT(A-C))                              DQG  41
      C=.49280575577263420D+0*B                                               DQG  42
      Y=Y+.81371973654528+D+2*(FCT(A+C)+FCT(A-C))                             DQG  43
      C=.48230112779375320D+0*B                                               DQG  44
      Y=Y+.12960325663103D+1*(FCT(A+C)+FCT(A-C))                              DQG  45
      C=.46745303796886980D+0*B                                               DQG  46
      Y=Y+.17136931456510720+1*(FCT(A+C)+FCT(A-C))                            DQG  47
      C=.44816057788302610D+0*B                                               DQG  48
      Y=Y+.21417949011113340+1*(FCT(A+C)+FCT(A-C))                            DQG  49
      C=.42468380686285000+0*B                                                DQG  50
      Y=Y+.25499029631188090+1*(FCT(A+C)+FCT(A-C))                            DQG  51
      C=.39724189798397120D+0*B                                               DQG  52
      Y=Y+.29342046739267770+1*(FCT(A+C)+FCT(A-C))                            DQG  53
      C=.36609105937014480D+0*B                                               DQG  54
      Y=Y+.32911111388180920+1*(FCT(A+C)+FCT(A-C))                            DQG  55
      C=.33152213346510760D+0*B                                               DQG  56
      Y=Y+.36172897054424250+1*(FCT(A+C)+FCT(A-C))                            DQG  57
      C=.29385787862038120D+0*B                                               DQG  58
      Y=Y+.39096947893535150+1*(FCT(A+C)+FCT(A-C))                            DQG  59
      C=.25344995446611470D+0*B                                               DQG  60
      Y=Y+.41655962113473380+1*(FCT(A+C)+FCT(A-C))                            DQG  61
      C=.21067563806531770D+0*B                                               DQG  62
      Y=Y+.43826046502201910+1*(FCT(A+C)+FCT(A-C))                            DQG  63
      C=.16593430114106380D+0*B                                               DQG  64
      Y=Y+.45586939347881900+1*(FCT(A+C)+FCT(A-C))                            DQG  65
      C=.11964368112606850D+0*B                                               DQG  66
      Y=Y+.46922199540402280+1*(FCT(A+C)+FCT(A-C))                            DQG  67
      C=.72235980791398200+1*B                                                DQG  68
      Y=Y+.47819360039637430+1*(FCT(A+C)+FCT(A-C))                            DQG  69
      C=.24153832843869160+1*B                                                DQG  70
      Y=B*(Y+.48270044257363900+1*(FCT(A+C)+FCT(A-C)))                        DQG  71
      RETURN                                                                  DQG  72
      END                                                                     DQG  73
      DOUBLE PRECISION FUNCTION HUB(X)                                        HUB   1
C     ***************************************************************       HUB   2
C                                                                             HUB   3
C     FUNCTION HUB                                                            HUB   4
C     PURPOSE                                                                 HUB   5
C        TO COMPUTE VALUES OF THE INTEGRAND OF H(U,B)                         HUB   6
C     DESCRIPTION OF PARAMETER                                                HUB   7
C        X - DOUBLE PRECISION, POINT AT WHICH INTEGRAND IS EVALUATED,         HUB   8
```

TABLE 5.2—*Listing of program for radial flow in a leaky artesian aquifer with storage of water in the confining beds*—Continued

```
C     METHOD                                                                    HUB   9
C       FORTRAN EVALUATION OF FUNCTION.                                         HUB  10
C                                                                               HUB  11
C     ***************************************************************************HUB  12
C                                                                               HUB  13
      IMPLICIT REAL*8(A-H,O-Z)                                                  HUB  14
      COMMON UUU,RRR                                                            HUB  15
      ARG=DSQRT((8RR+8RR+UUU)/(X+X+X+UUU))                                      HUB  16
      HUB=DEXP(-X)*DERFC(ARG)/X                                                 HUB  17
      RETURN                                                                    HUB  18-
      END
      DOUBLE PRECISION FUNCTION W(U)                                            WU   1
C     ***************************************************************************WU   2
C                                                                               WU   3
C     FUNCTION W                                                                WU   4
C     PURPOSE                                                                   WU   5
C        TO EVALUATE THE WELL FUNCTION OF THEIS.                                WU   6
C     DESCRIPTION OF PARAMETER                                                  WU   7
C        U = DOUBLE PRECISION, ARGUMENT FOR WELL FUNCTION.                      WU   8
C                                                                               WU   9
C     ***************************************************************************WU  10
      IMPLICIT REAL*8 (A-H,O-Z)                                                 WU  11
      IF (U.LE.0.0) GO TO 2                                                     WU  12
      IF (U.GT.100.) GO TO 3                                                    WU  13
      IF (U.GE.1.0) GO TO 1                                                     WU  14
      W=-.57721566+U*(.99999193+U*(-.24991055+U*(.05519968+U*(-.00976004 WU  15
     1+.00107857*U))))-DLOG(U)                                                  WU  16
      GO TO 4                                                                   WU  17
    1 ENUM=DEXP(-U)*(.2677737343+U*(8.6347608925+U*(18.059016973+U*(8.5 WU  18
     1733287401+U))))                                                           WU  19
      DEN=U*(3.9584969228+U*(21.0996530827+U*(25.6329561486+U*(9.5733223 WU  20
     1654+U))))                                                                 WU  21
      W=ENUM/DEN                                                                WU  22
      GO TO 4                                                                   WU  23
    2 WRITE (6,5) U                                                             WU  24
      STOP                                                                      WU  25
    3 W=0.0                                                                     WU  26
    4 RETURN                                                                    WU  27
C                                                                               WU  28
    5 FORMAT ('0',5X,'W(U) NOT DEFINED FOR U=',1PD15.8)                         WU  29
      END                                                                       WU  30-
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*

```
C     ***************************************************************************PPL   1
C                                                                                 PPL   2
C                                                                                 PPL   3
C     PURPOSE                                                                     PPL   4
C        TO COMPUTE TYPE CURVE FUNCTION VALUES FOR PARTIAL PENETRATION            PPL   5
C        IN A LEAKY AQUIFER USING EQ. 73 OF HANTUSH,M.S., 1964,                   PPL   6
C        HYDRAULICS OF WELLS IN CHOW, VEN TE, ADVANCES IN HYDROSCIENCE,           PPL   7
C        VOL. 1: ACADEMIC PRESS, NEW YORK, P. 281-442.                            PPL   8
C     INPUT DATA                                                                  PPL   9
C        1 CARD  =  FORMAT (3F5.1,I5,2E10.4)                                      PPL  10
C             B = AQUIFER THICKNESS                                               PPL  11
C             E = DEPTH, BELOW TOP OF AQUIFER, TO BOTTOM OF PUMPING               PPL  12
C             WELL SCREEN                                                         PPL  13
C             D = DEPTH, BELOW TOP OF AQUIFER, TO TOP OF PUMPING WELL             PPL  14
C             SCREEN                                                              PPL  15
C             NUM = NUMBER OF OBSERVATION WELLS OR PIEZOMETERS TIMES              PPL  16
C             NUMBER OF VALUES OF KZ/KR.                                          PPL  17
C             SMALL = SMALLEST VALUE OF 1/U FOR WHICH COMPUTATION IS              PPL  18
C             DESIRED                                                             PPL  19
C             LARGE = LARGEST  VALUE OF 1/U FOR WHICH COMPUTATION IS              PPL  20
C             DESIRED                                                             PPL  21
C        2 CARDS = FORMAT(8E10.5)                                                 PPL  22
C             BOAT = 12 VALUES OF R/BR, NON ZERO VALUES SHOULD BE                 PPL  23
C             FIRST, WILL TERMINATE AT FIRST ZERO (OR BLANK) VALUE.               PPL  24
C        NUM CARDS (ONE FOR EACH OBS. WELL OR PIEZOMETER AND FOR EACH             PPL  25
C             VALUE OF R*SQRT(KZ/KR) = FORMAT (3F5.1)                             PPL  26
C             R = RADIAL DISTANCE FROM PUMPED WELL TIMES SQRT(KZ/KR),             PPL  27
C             LPRIME = DEPTH, BELOW TOP OF AQUIFER, TO BOTTOM OF OBS.             PPL  28
C             WELL SCREEN (ZERO FOR PIEZOMETER)                                   PPL  29
C             DPRIME = DEPTH, BELOW TOP OF AQUIFER, TO TOP OF OBS. WELL
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
C            SCREEN (TOTAL DEPTH FOR PIEZOMETER)                                PPL  30
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                             PPL  31
C        DGL12,SERIES,BESK,FCT,L,FL                                             PPL  32
C                                                                               PPL  33
C     ***************************************************************************PPL  34
      REAL*8 U,V                                                                PPL  35
      REAL*4 L,LB,LPB,LPRIME,LARGE                                              PPL  36
      DIMENSION ARRAY(55,12), ARG(6), BDAT(12), Y(55)                           PPL  37
      DATA ARG/1.,1.5,2.,3.,5.,7./                                              PPL  38
      DATA ARRAY/660*0./,Y/55*0./                                               PPL  39
      IRD=5                                                                     PPL  40
      IPT=6                                                                     PPL  41
      READ (IRD,9) B,E,D,NUM,SMALL,LARGE                                        PPL  42
      READ (IRD,14) BDAT                                                        PPL  43
      DO 1 I=1,12                                                               PPL  44
      IF (BDAT(I).EQ.0.) GO TO 2                                                PPL  45
    1 CONTINUE                                                                  PPL  46
      NB=12                                                                     PPL  47
      GO TO 3                                                                   PPL  48
    2 NB=I-1                                                                    PPL  49
    3 LB=E/B                                                                    PPL  50
      DB=D/B                                                                    PPL  51
      IBEGIN=ALOG10(SMALL)                                                      PPL  52
      IEND=ALOG10(LARGE)+.1                                                     PPL  53
      JLIMIT=IEND-IBEGIN                                                        PPL  54
      IF (JLIMIT.GT.9) JLIMIT=9                                                 PPL  55
      ILIMIT=6*JLIMIT+1                                                         PPL  56
      DO 8 K=1,NUM                                                              PPL  57
      READ (IRD,9) R,LPRIME,DPRIME                                              PPL  58
      RB=R/B                                                                    PPL  59
      LPB=LPRIME/B                                                              PPL  60
      DPB=DPRIME/B                                                              PPL  61
      DO 4 I=1,ILIMIT                                                           PPL  62
      INDEX=(I-1)/6                                                             PPL  63
      IEXP=IBEGIN+INDEX                                                         PPL  64
      II=I-INDEX*6                                                              PPL  65
      Y(I)=ARG(II)*10.**IEXP                                                    PPL  66
      U=1./Y(I)                                                                 PPL  67
      DO 4 J=1,NB                                                               PPL  68
      BETA=BDAT(J)                                                              PPL  69
      V=BETA/2.                                                                 PPL  70
    4 ARRAY(I,J)=L(U,V)+FL(U,RB,BETA,LB,DB,LPB,DPB)                             PPL  71
      IF (LPB=0.) 5,5,6                                                         PPL  72
    5 WRITE (IPT,10) DPB,NB,LB,DB                                               PPL  73
      GO TO 7                                                                   PPL  74
    6 WRITE (IPT,11) LPB,DPB,RB,LB,DB                                           PPL  75
    7 WRITE (IPT,12) (BDAT(I),I=1,NB)                                           PPL  76
      DO 8 I=1,ILIMIT                                                           PPL  77
      WRITE (IPT,13) Y(I),(ARRAY(I,J),J=1,NB)                                   PPL  78
    8 CONTINUE                                                                  PPL  79
      STOP                                                                      PPL  80
C                                                                               PPL  81
C                                                                               PPL  82
    9 FORMAT (3F5.1,I5,2E10.4)                                                  PPL  83
   10 FORMAT ('1','W(U,R/B,R)*F(U,R/B,R/B,L/B,D/B,Z/B), Z/B=',F5.2,', SQPPL  84
     1RT(KZ/KR)*R/B=',F5.2,', L/B=',F5.2,', D/B=',F5.2)                         PPL  85
   11 FORMAT ('1','W(U,R/B,R)*F(U,R/B,R/B,L/B,D/B,L''/B,D''/B), L''/B=',PPL  86
     1F5.2,', D''/B=',F5.2,', SQRT(KZ/KR)*R/B=',F5.2,', L/B=',F5.2,', D/PPL  87
     2B=',F5.2)                                                                 PPL  88
   12 FORMAT ('0',9X,'1 R/B','/' ',5X,'1/U ',12E10.2)                           PPL  89
   13 FORMAT (' ',E10.3,12F10.4)                                                PPL  90
   14 FORMAT (8E10.5)                                                           PPL  91
      END                                                                       PPL  92=
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
      REAL FUNCTION FL*4(U,RB,BETA,LB,DB,LPB,DPB)                                FL   1
C     ***************************************************************************FL   2
C                                                                                FL   3
C                                                                                FL   4
C     FUNCTION FL                                                                FL   5
C                                                                                FL   6
C     PURPOSE                                                                    FL   7
C        TO COMPUTE DEPARTURES FROM HANTUSH-JACOB LEAKY AQUIFER CURVE            FL   8
C        CAUSED BY PARTIAL PENETRATION OF PUMPED WELL.                           FL   9
C     USAGE                                                                      FL  10
C        FL(U,RB,BETA,LB,DB,LPB,DPB)                                             FL  11
C     DESCRIPTION OF PARAMETERS                                                  FL  12
C        ALL REAL, U DOUBLE PRECISION                                            FL  13
C        U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE                  FL  14
C            COEFFICIENT / 4*TRANSMISSIVITY * TIME                               FL  15
C        RB = R/B ( RADIAL DISTANCE / AQUIFER THICKNESS )                        FL  16
C        BETA = R*SQRT(K'/B'T) = (RADIAL DISTANCE * SQUARE ROOT                  FL  17
C               (HYD. COND. OF CONFINING BED/THICKNESS OF CONFINING              FL  18
C               BED * TRANSMISSIVITY OF AQUIFER))                                FL  19
C        LB = L/B ( FRACTION OF AQUIFER PENETRATED BY PUMPED WELL)               FL  20
C        DB = D/B ( FRACTION OF AQUIFER ABOVE PUMPED WELL SCREEN)                FL  21
C        LPB = L'/B (FRACTION OF AQUIFER PENETRATED BY OBS. WELL, ZERO           FL  22
C              FOR PIEZOMETER)                                                   FL  23
C        DPB = D'/B (FRACTION OF AQUIFER ABOVE OBS. WELL SCREEN, TOTAL           FL  24
C              DEPTH FOR PIEZOMETER)                                             FL  25
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                              FL  26
C        DQL12,SERIES,BESK,FCT,L                                                 FL  27
C     METHOD                                                                     FL  28
C        SUMS THE SERIES THROUGH N*PI*R/B EQ 20                                  FL  29
C                                                                                FL  30
C     ***************************************************************************FL  31
      REAL*8 U,V,DSQRT                                                           FL  32
      REAL*4 L,N,LB,LPB                                                          FL  33
      SUM=0.                                                                     FL  34
      N=0.                                                                       FL  35
      BETSQ=BETA*BETA                                                            FL  36
      PIRBSQ=9.869604*RB*RB                                                      FL  37
      PILB=3.141593*LB                                                           FL  38
      PIDB=3.141593*DB                                                           FL  39
      IF (LPB=0.) 1,1,4                                                          FL  40
C     CHECKS FOR WELL OR PIEZOMETER                                              FL  41
    1 PIZB=3.141593*DPB                                                          FL  42
    2 N=N+1.                                                                     FL  43
      V=SQRT(BETSQ+N*N*PIRBSQ)/2.                                                FL  44
      IF (V.GT.10.) GO TO 3                                                      FL  45
C     TRUNCATES SERIES WHEN V>10                                                 FL  46
      X=L(U,V)/N                                                                 FL  47
      SUM=SUM+(SIN(N*PILB)-SIN(N*PIDB))*COS(N*PIZB)*X                            FL  48
      GO TO 2                                                                    FL  49
    3 FL=.6366198*SUM/(LB-DB)                                                    FL  50
      GO TO 7                                                                    FL  51
    4 PILPB=3.141593*LPB                                                         FL  52
      PIDPB=3.141593*DPB                                                         FL  53
    5 N=N+1                                                                      FL  54
      V=SQRT(BETSQ+N*N*PIRBSQ)/2.                                                FL  55
      IF (V.GT.10.) GO TO 6                                                      FL  56
C     TRUNCATES SERIES WHEN V>10                                                 FL  57
      X=L(U,V)/N                                                                 FL  58
      SUM=SUM+(SIN(N*PILB)-SIN(N*PIDB))*(SIN(N*PILPB)-SIN(N*PIDPB))*X/N          FL  59
      GO TO 5                                                                    FL  60
    6 FL=.2026424*SUM/((LB-DB)*(LPB-DPB))                                        FL  61
    7 RETURN                                                                     FL  62
      END
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
      REAL FUNCTION L*4(U,V)                                                L   1
C     ***********************************************************************L   2
C                                                                           L   3
C     FUNCTION L                                                            L   4
C                                                                           L   5
C     PURPOSE                                                               L   6
C        TO COMPUTE THE INTEGRAL( EXP(-Y-V**2/Y)/Y) SUMMED OVER Y FROM      L   7
C        U TO INFINITY(WELL FUNCTION FOR LEAKY AQUIFERS).                   L   8
C     DESCRIPTION OF PARAMETERS                                             L   9
C        BOTH DOUBLE PRECISION                                              L  10
C        U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE             L  11
C            COEFFICIENT / 4*TRANSMISSIVITY * TIME                          L  12
C        V = R/2*SQRT(K'/(T*B'))--ONE-HALF RADIAL DISTANCE*SQUARE ROOT      L  13
C            (HYD. COND. OF CONFINING BED/TRANSMISSIVITY*THICKNESS          L  14
C            OF CONFINING BED)                                              L  15
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                         L  16
C        DQL12,SERIES,BESK,FCT                                              L  17
C     METHOD                                                                L  18
C        IN THE FOLLOWING F=EXP(-Y-V**2/Y)/Y                                L  19
C      (1) U>=1, USES A GAUSSIAN-LAGUERRE QUADRATURE FORMULA TO             L  20
C        EVALUATE INTEGRAL(F) FROM U TO INF.                                L  21
C      (2) V**2<U<1, USES THE G-L QUADRATURE TO EVALUATE INTEGRAL(F)        L  22
C        FROM ONE TO INF AND A SERIES EXPANSION TO EVALUATE INTEGRAL(F)     L  23
C        FROM U TO ONE.                                                     L  24
C      (3) U<1, U<=V**2, USES THE REPRESENTATION INTEGRAL(F) FROM U         L  25
C        TO INF. = 2*K0(2*V)-INTEGRAL(F) FROM V**2/U TO INF.                L  26
C        EVALUATES THE ZERO ORDER MODIFIED BESSEL FUNCTION OF SECOND        L  27
C        KIND WITH IBM SUBROUTINE, EVALUATES INTEGRAL BY G-L QUAD.          L  28
C                                                                           L  29
C     ***********************************************************************L  30
      EXTERNAL FCT                                                          L  31
      REAL*8 U,V,Z,F,VV,SERIES                                              L  32
      COMMON /C1/ VV,Z                                                      L  33
      VV=V                                                                  L  34
      IF (U-1.) 1,2,2                                                       L  35
C     CHECKS IF U<1                                                         L  36
    1 Z=V*V/U                                                               L  37
      IF (Z-1.) 3,4,4                                                       L  38
C     CHECKS IF V**2/U < 1                                                  L  39
    2 Z=U                                                                   L  40
      CALL DQL12(FCT,F)                                                     L  41
      L=F                                                                   L  42
C     INTEGRAL U TO INF, EVALUATED BY GAUSS-LAGUERRE QUADRATURE             L  43
      GO TO 5                                                               L  44
    3 Z=1.                                                                  L  45
      CALL DQL12(FCT,F)                                                     L  46
      L=F+SERIES(U,V)                                                       L  47
C     INTEGRAL 1 TO INF, BY G-L QUAD., INTEGRAL U TO 1 BY SERIES EXP.       L  48
      GO TO 5                                                               L  49
    4 TWOV=2.*V                                                             L  50
      CALL BESK(TWOV,0,BK,IER)                                              L  51
      CALL DQL12(FCT,F)                                                     L  52
      L=2.*BK-F                                                             L  53
C     2K0(2V)-INTEGRAL V**2/U TO INF.                                       L  54
    5 RETURN                                                                L  55
      END                                                                   L  56-
      REAL FUNCTION SERIES*8(U,V)                                         SER   1
C     ***********************************************************************SER   2
C                                                                         SER   3
C     FUNCTION SERIES                                                     SER   4
C                                                                         SER   5
C     PURPOSE                                                             SER   6
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
C     TO EVALUATE S(1)=S(U), WHERE S IS A SERIES EXPANSION OF                SER   7
C     INTEGRAL(EXP(-Y-V**2/Y)DY/Y) GIVEN BY: S= SUM, M=0 TO INFINITY,        SER   8
C     (F(M)*SUM, N=0 TO INF.,(V**(2*N)/((N!)*(M+N)!)) WHERE F(M)=            SER   9
C     LOG(U) IF M=0 AND = ((-1)**M/M)*(U**M-(V**2/U)**M) IF M>0.             SER  10
C                                                                            SER  11
C   DESCRIPTION OF PARAMETERS                                                SER  12
C     BOTH DOUBLE PRECISION                                                  SER  13
C     U = R**2*S/4*T*TIME (RADIAL DISTANCE SQUARED * STORAGE                 SER  14
C         COEFFICIENT / 4*TRANSMISSIVITY * TIME                              SER  15
C     V = R/2*SQRT(K'/(T*B'))--ONE-HALF RADIAL DISTANCE*SQUARE ROOT          SER  16
C         (HYD. COND. OF CONFINING BED/TRANSMISSIVITY*THICKNESS              SER  17
C         OF CONFINING BED)                                                  SER  18
C   SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                            SER  19
C     NONE                                                                   SER  20
C   METHOD                                                                   SER  21
C     SUMMATION IS TERMINATED FOR THE INNER SERIES WHEN A TERM               SER  22
C     BECOMES LESS THAN 5.E-7/N AND FOR OUTER SERIES WHEN A TERM             SER  23
C     BECOMES LESS THAN 5.E-7                                                SER  24
C                                                                            SER  25
C     ***************************************************************      SER  26
      REAL*8 DLOG,DABS,S(2),VUM,UU                                           SER  27
      REAL*8 TEST,U,UM,EM,EN,SUM1,SUM,SIGN,V,VSQ,VSQU,RMUL,TERM,TERM1        SER  28
      TEST=5.D-07                                                            SER  29
      VSQ=V*V                                                                SER  30
      UU=U                                                                   SER  31
      DO 6 I=1,2                                                             SER  32
C     EVALUATES SERIES FOR LOWER LIMIT = U AND UPPER LIMIT = 1               SER  33
      IF (I.EQ.2) U=1.                                                       SER  34
      UM=1.                                                                  SER  35
      EM=-1.                                                                 SER  36
      SUM1=0.                                                                SER  37
      SIGN=-1.                                                               SER  38
      VUM=1.                                                                 SER  39
      VSQU=VSQ/U                                                             SER  40
    1 EM=EM+1.                                                               SER  41
      IF (EM-.1) 2,3,3                                                       SER  42
C     CHECKS FOR M=0                                                         SER  43
    2 RMUL=DLOG(U)                                                           SER  44
      TERM1=1.                                                               SER  45
      GO TO 4                                                                SER  46
    3 UM=UM*U                                                                SER  47
      IF (VUM.LT.1.D-30) VUM=0.                                              SER  48
      VUM=VUM*VSQU                                                           SER  49
      RMUL=(UM-VUM)/EM                                                       SER  50
      TERM1=TERM1/EM                                                         SER  51
    4 SIGN=-SIGN                                                             SER  52
      SUM=TERM1                                                              SER  53
      TERM=TERM1                                                             SER  54
      EN=0.                                                                  SER  55
    5 EN=EN+1.                                                               SER  56
      TERM=TERM*VSQ/(EN*(EN+EM))                                             SER  57
      SUM=SUM+TERM                                                           SER  58
      IF (TEST.LE.DABS(RMUL*EN*TERM)) GO TO 5                                SER  59
C     TRUNCATES INNER SERIES IF OUTER TERM*N*INNER TERM < 5.E-7               SER  60
      SUM1=SUM1+SIGN*RMUL*SUM                                                SER  61
      IF (EM.LT.-.1) GO TO 1                                                 SER  62
      IF (TEST.LE.DABS(RMUL*SUM)) GO TO 1                                    SER  63
C     TRUNCATES OUTER SERIES IF OUTER TERM*INNER SUM < 5.E-7                  SER  64
    6 S(I)=SUM1                                                              SER  65
      U=UU                                                                   SER  66
      SERIES=S(2)-S(1)                                                       SER  67
      RETURN                                                                 SER  68-
      END
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
      REAL FUNCTION FCT*8(X)                                                FCT    1
C     ********************************************************************FCT    2
C                                                                           FCT    3
C     FUNCTION FCT                                                          FCT    4
C                                                                           FCT    5
C     PURPOSE                                                               FCT    6
C        TO COMPUTE FCT(X)=EXP(-Z-V**2/(X+Z))/(X+Z)                         FCT    7
C     DESCRIPTION OF PARAMETERS                                             FCT    8
C        X = THE DOUBLE PRECISION VALUE OF X FOR WHICH FCT IS COMPUTED      FCT    9
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                         FCT   10
C        NONE                                                               FCT   11
C     METHOD                                                                FCT   12
C        FORTRAN EVALUATION OF FUNCTION                                     FCT   13
C                                                                           FCT   14
C     ********************************************************************FCT   15
      REAL*8 X,V,Z,P,DEXP                                                   FCT   16
      COMMON /C1/ V,Z                                                       FCT   17
      IF (X) 1,2,2                                                          FCT   18
    1 FCT=0.                                                                FCT   19
      GO TO 4                                                               FCT   20
    2 P=Z+V**2/(X+Z)                                                        FCT   21
      IF (P-5.D1) 3,3,1                                                     FCT   22
    3 FCT=DEXP(-P)/(X+Z)                                                    FCT   23
    4 RETURN                                                                FCT   24
      END                                                                   FCT   25-
      SUBROUTINE DQL12(FCT,Y)                                               DL12  380
C                                                                           DL12   10
C     ********************************************************************DL12   20
C                                                                           DL12   30
C        SUBROUTINE DQL12                                                   DL12   40
C                                                                           DL12   50
C        PURPOSE                                                            DL12   60
C           TO COMPUTE INTEGRAL(EXP(-X)*FCT(X), SUMMED OVER X               DL12   70
C                              FROM 0 TO INFINITY),                         DL12   80
C                                                                           DL12   90
C        USAGE                                                              DL12  100
C           CALL DQL12 (FCT,Y)                                              DL12  110
C           PARAMETER FCT REQUIRES AN EXTERNAL STATEMENT                    DL12  120
C                                                                           DL12  130
C        DESCRIPTION OF PARAMETERS                                          DL12  140
C           FCT    - THE NAME OF AN EXTERNAL DOUBLE PRECISION FUNCTION      DL12  150
C                    SUBPROGRAM USED,                                       DL12  160
C           Y      - THE RESULTING DOUBLE PRECISION INTEGRAL VALUE,         DL12  170
C                                                                           DL12  180
C        REMARKS                                                            DL12  190
C           NONE                                                            DL12  200
C                                                                           DL12  210
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                      DL12  220
C           THE EXTERNAL DOUBLE PRECISION FUNCTION SUBPROGRAM FCT(X)        DL12  230
C           MUST BE FURNISHED BY THE USER,                                  DL12  240
C                                                                           DL12  250
C        METHOD                                                             DL12  260
C           EVALUATION IS DONE BY MEANS OF 12-POINT GAUSSIAN-LAGUERRE       DL12  270
C           QUADRATURE FORMULA, WHICH INTEGRATES EXACTLY,                   DL12  280
C           WHENEVER FCT(X) IS A POLYNOMIAL UP TO DEGREE 23,                DL12  290
C           FOR REFERENCE, SEE                                              DL12  300
C           SHAO/CHEN/FRANK, TABLES OF ZEROS AND GAUSSIAN WEIGHTS OF        DL12  310
C           CERTAIN ASSOCIATED LAGUERRE POLYNOMIALS AND THE RELATED         DL12  320
C           GENERALIZED HERMITE POLYNOMIALS, IBM TECHNICAL REPORT           DL12  330
C           TR00.1100 (MARCH 1964), PP.24-25.                               DL12  340
C                                                                           DL12   50
C     ********************************************************************DL12  360
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
C                                                                       DL12 370
C                                                                       DL12 390
C                                                                       DL12 400
C                                                                       DL12 410
      DOUBLE PRECISION X,Y,FCT                                          DL12 420
C                                                                       DL12 430
      X=.3709912104446692 D2                                            DL12 440
      Y=.8148077467426240 D-15*FCT(X)                                   DL12 450
      X=.2848796725098400 D2                                            DL12 460
      Y=Y+.3061601635035021 D-11*FCT(X)                                 DL12 470
      X=.2215109037939701 D2                                            DL12 480
      Y=Y+.1342391030515004 D-8*FCT(X)                                  DL12 490
      X=.1711685516746226 D2                                            DL12 500
      Y=Y+.1668493876540910 D-6*FCT(X)                                  DL12 510
      X=.1300605499330635 D2                                            DL12 520
      Y=Y+.8365055856681980 D-5*FCT(X)                                  DL12 530
      X=.9621316842456870 D1                                            DL12 540
      Y=Y+.2032315926629994 D-3*FCT(X)                                  DL12 550
      X=.6844525453115177 D1                                            DL12 560
      Y=Y+.2663973541865316 D-2*FCT(X)                                  DL12 570
      X=.4599227639418348 D1                                            DL12 580
      Y=Y+.2010238115463410 D-1*FCT(X)                                  DL12 590
      X=.2833751337743507 D1                                            DL12 600
      Y=Y+.9044922221168090 D-1*FCT(X)                                  DL12 610
      X=.1512610269776419 D1                                            DL12 620
      Y=Y+.2440820113198776 D0*FCT(X)                                   DL12 630
      X=.6117574845151307 D0                                            DL12 640
      Y=Y+.3777592758731380 D0*FCT(X)                                   DL12 650
      X=.1157221173580207 D0                                            DL12 660
      Y=Y+.2647313710554432 D0*FCT(X)                                   DL12 670
      RETURN                                                            DL12 680
      END                                                               BESK 410
      SUBROUTINE BESK(X,N,BK,IER)                                       BESK  10
C                                                                       BESK  20
C     ..................................................................BESK  30
C                                                                       BESK  40
C        SUBROUTINE BESK                                                BESK  50
C                                                                       BESK  60
C           COMPUTE THE K BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDERBESK  70
C                                                                       BESK  80
C        USAGE                                                          BESK  90
C           CALL BESK(X,N,BK,IER)                                       BESK 100
C                                                                       BESK 110
C        DESCRIPTION OF PARAMETERS                                      BESK 120
C           X  =THE ARGUMENT OF THE K BESSEL FUNCTION DESIRED           BESK 130
C           N  =THE ORDER OF THE K BESSEL FUNCTION DESIRED              BESK 140
C           BK =THE RESULTANT K BESSEL FUNCTION                         BESK 150
C           IER=RESULTANT ERROR CODE WHERE                              BESK 160
C              IER=0  NO ERROR                                          BESK 170
C              IER=1  N IS NEGATIVE                                     BESK 180
C              IER=2  X IS ZERO OR NEGATIVE                             BESK 190
C              IER=3  X .GT. 170, MACHINE RANGE EXCEEDED                BESK 200
C              IER=4  BK .GT. 10**70                                    BESK 210
C                                                                       BESK 220
C        REMARKS                                                        BESK 230
C           N MUST BE GREATER THAN OR EQUAL TO ZERO                     BESK 240
C                                                                       BESK 250
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                  BESK 260
C           NONE                                                        BESK 270
C                                                                       BESK 280
C        METHOD                                                         BESK 290
C           COMPUTES ZERO ORDER AND FIRST ORDER BESSEL FUNCTIONS USING  BESK 300
C           SERIES APPROXIMATIONS AND THEN COMPUTES N TH ORDER FUNCTION BESK 310
C           USING RECURRENCE RELATION.
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
C       RECURRENCE RELATION AND POLYNOMIAL APPROXIMATION TECHNIQUE   BESK 320
C       AS DESCRIBED BY A.J.M.HITCHCOCK,'POLYNOMIAL APPROXIMATIONS   BESK 330
C       TO BESSEL FUNCTIONS OF ORDER ZERO AND ONE AND TO RELATED     BESK 340
C       FUNCTIONS', M.T.A.C., V.11,1957,PP.86-88, AND G.N. WATSON,   BESK 350
C       'A TREATISE ON THE THEORY OF BESSEL FUNCTIONS', CAMBRIDGE    BESK 360
C       UNIVERSITY PRESS, 1958, P. 62                                BESK 370
C                                                                    BESK 380
C     ..............................................................BESK 390
C                                                                    BESK 400
      DIMENSION T(12)                                                BESK 420
      BK=.0                                                          BESK 430
      IF(N)10,11,11                                                  BESK 440
   10 IER=1                                                          BESK 450
      RETURN                                                         BESK 460
   11 IF(X)12,12,20                                                  BESK 470
   12 IER=2                                                          BESK 480
      RETURN                                                         BESK 490
   20 IF(X-170.0)22,22,21                                            BESK 500
   21 IER=3                                                          BESK 510
      RETURN                                                         BESK 520
   22 IER=0                                                          BESK 530
      IF(X-1.)36,36,25                                               BESK 540
   25 A=EXP(-X)                                                      BESK 550
      B=1./X                                                         BESK 560
      C=SQRT(B)                                                      BESK 570
      T(1)=B                                                         BESK 580
      DO 26 L=2,12                                                   BESK 590
   26 T(L)=T(L-1)*B                                                  BESK 600
      IF(N-1)27,29,27                                                BESK 610
C                                                                    BESK 620
C     COMPUTE KO USING POLYNOMIAL APPROXIMATION                      BESK 630
C                                                                    BESK 640
   27 G0=A*(1.2533141-.1566642*T(1)+.08811128*T(2)-.09139095*T(3)    BESK 650
     2+.1344596*T(4)-.2499850*T(5)+.3792410*T(6)-.5247277*T(7)       BESK 660
     3+.5575368*T(8)-.4262633*T(9)+.2184518*T(10)-.06680977*T(11)    BESK 670
     4+.009189383*T(12))*C                                           BESK 680
      IF(N)20,28,29                                                  BESK 690
   28 BK=G0                                                          BESK 700
      RETURN                                                         BESK 710
C                                                                    BESK 720
C     COMPUTE K1 USING POLYNOMIAL APPROXIMATION                      BESK 730
C                                                                    BESK 740
   29 G1=A*(1.2533141+.4699927*T(1)-.1468583*T(2)+.1280427*T(3)      BESK 750
     2-.1736432*T(4)+.2847618*T(5)-.4594342*T(6)+.6283381*T(7)       BESK 760
     3-.6632295*T(8)+.5050239*T(9)-.2581304*T(10)+.07880001*T(11)    BESK 770
     4-.01062418*T(12))*C                                            BESK 780
      IF(N-1)20,30,31                                                BESK 790
   30 BK=G1                                                          BESK 800
      RETURN                                                         BESK 810
C                                                                    BESK 820
C     FROM KO,K1 COMPUTE KN USING RECURRENCE RELATION                BESK 830
C                                                                    BESK 840
   31 DO 35 J=2,N                                                    BESK 850
      GJ=2.*(FLOAT(J)-1.)*G1/X+G0                                    BESK 860
      IF(GJ-1.0E70)33,33,32                                          BESK 870
   32 IER=4                                                          BESK 880
      GO TO 34                                                       BESK 890
   33 G0=G1                                                          BESK 900
   35 G1=GJ                                                          BESK 910
   34 BK=GJ                                                          BESK 920
      RETURN                                                         BESK 930
   36 B=X/2.                                                         BESK 940
```

TABLE 6.1.—*Listing of program for partial penetration in a leaky artesian aquifer*—Continued

```
      A=.5772157+ALOG(B)                                                BESK 950
      C=B*B                                                             BESK 960
      IF(N-1)37,43,37                                                   BESK 970
C                                                                       BESK 980
C     COMPUTE K0 USING SERIES EXPANSION                                 BESK 990
C                                                                       BESK1000
   37 G0=-A                                                             BESK1010
      X2J=1.                                                            BESK1020
      FACT=1.                                                           BESK1030
      HJ=.0                                                             BESK1040
      DO 40 J=1,6                                                       BESK1050
      RJ=1./FLOAT(J)                                                    BESK1060
      IF(X2J.LT.1.E-40) X2J=0.                                          BESK1061
C     PREVIOUS STATEMENT ADDED TO IBM SUBROUTINE TO CORRECT UNDERFLOW   BESK1062
C     PROBLEM ON WATFOR COMPILER                                        BESK1063
      X2J=X2J*C                                                         BESK1070
      FACT=FACT*RJ*RJ                                                   BESK1080
      HJ=HJ+RJ                                                          BESK1090
   40 G0=G0+X2J*FACT*(HJ-A)                                             BESK1100
      IF(N)43,42,43                                                     BESK1110
   42 BK=G0                                                             BESK1120
      RETURN                                                            BESK1130
C                                                                       BESK1140
C     COMPUTE K1 USING SERIES EXPANSION                                 BESK1150
C                                                                       BESK1160
   43 X2J=B                                                             BESK1170
      FACT=1.                                                           BESK1180
      HJ=1.                                                             BESK1190
      G1=1./X+X2J*(.5+A-HJ)                                             BESK1200
      DO 50 J=2,8                                                       BESK1210
      X2J=X2J*C                                                         BESK1220
      RJ=1./FLOAT(J)                                                    BESK1230
      FACT=FACT*RJ*RJ                                                   BESK1240
      HJ=HJ+RJ                                                          BESK1250
   50 G1=G1+X2J*FACT*(.5+(A-HJ)*FLOAT(J))                               BESK1260
      IF(N-1)51,52,51                                                   BESK1270
   52 BK=G1                                                             BESK1280
      RETURN                                                            BESK1290
      END                                                               BESK130-
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*

```
C     ********************************************************************* Z   1
C                                                                           Z   2
C                                                                           Z   3
C     PURPOSE                                                               Z   4
C        TO COMPUTE A TABLE OF FUNCTION VALUES FOR DRAWDOWN IN A            Z   5
C        LEAKY ARTESIAN AQUIFER IN RESPONSE TO A STEP CHANGE IN             Z   6
C        WATER LEVEL IN THE CONTROL WELL, FUNCTION VALUES ARE               Z   7
C        EXPRESSED AS A FRACTION OF DRAWDOWN IN CONTROL WELL (S/Sw),        Z   8
C        REFERENCE - HANTUSH,M.S., 1959, NONSTEADY FLOW TO FLOWING          Z   9
C        WELLS IN LEAKY AQUIFERS: JOUR. GEOPHYS. RESEARCH, V. 64,           Z  10
C        NO. 8, P. 1043-1052.                                               Z  11
C     INPUT DATA                                                            Z  12
C        1 CARD - FORMAT(2E10.5)                                            Z  13
C             TSMALL - SMALLEST VALUE OF ALPHA FOR WHICH COMPUTATION        Z  14
C             IS DESIRED.                                                   Z  15
C             TLARGE -  LARGEST VALUE OF ALPHA FOR WHICH COMPUTATION        Z  16
C             IS DESIRED.                                                   Z  17
C        1 CARD - FORMAT(13F5.0)                                            Z  18
C             BDAT - 13 VALUES OF Rw/B, NON ZERO VALUES SHOULD BE GE 1      Z  19
C             AND LT 10. FIRST ZERO (OR BLANK) WILL TERMINATE THE           Z  20
C             LIST. AT LEAST ONE NON ZERO VALUE MUST BE CODED. INPUT        Z  21
C             VALUES ARE MULTIPLIED BY POWER OF TEN DETERMINED BY           Z  22
C             PROGRAM FROM ALPHA.
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
C      1 CARD = FORMAT(10F8.2)                                              Z  23
C           RW = RADIUS OF CONTROL WELL.                                    Z  24
C           RDAT = 9 VALUES OF RADIAL DISTANCE OF OBSERVATION POINTS        Z  25
C           FROM CONTROL WELL, SHOULD BE CODED WITH SMALLEST NUMBER         Z  26
C           FIRST, THEN BY INCREASING DISTANCE. THE FIRST ZERO              Z  27
C           (OR BLANK) VALUE WILL TERMINATE COMPUTATION.                    Z  28
C     METHOD                                                                Z  29
C       EVALUATES EQ. 13 OF MANTUSH. EVALUATION OF BESSEL FUNCTIONS         Z  30
C       BY SUBROUTINES BESK AND BESY AND FUNCTION J0. EVALUATES             Z  31
C       INTEGRAL BY SUM, I=1 TO 8000, F((DELTA U)*(I-.5))*(DELTA U).        Z  32
C       CHOOSES INITIAL DELTA U = .001/SQRT(SMALLEST ALPHA) AND USES        Z  33
C       THIS VALUE FOR ALL RW/B GE 10*(DELTA U). FOR SMALLER RW/B,          Z  34
C       DIVIDES DELTA U BY 10 AND MULTIPLIES SMALLEST ALPHA BY 100.         Z  35
C     REMARKS                                                               Z  36
C       SMALLEST RW/B GE .01/SQRT(SMALLEST ALPHA)                           Z  37
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                         Z  38
C       BESK,BESY,J0                                                        Z  39
C                                                                           Z  40
C     ***************************************************************       Z  41
      REAL*8 SUM1,SUM2                                                      Z  42
      REAL*4 KOBP,KOB,J0,J0PU,J0U,Y(8000),J(8000),F(8000),FT(8000),         Z  43
     1   FB(8000),RDAT(9),TDAT(6),BDAT(13),ARRAY(25,9,13),B(13),T(25)       Z  44
      DATA FT/8000*0./,FB/8000*0./                                          Z  45
      DATA RDAT/9*1./                                                       Z  46
      DATA ARRAY/2925*0./,TDAT/1.,1.5,2.,3.,5.,7./                          Z  47
      IRD=5                                                                 Z  48
      IPT=6                                                                 Z  49
      READ (IRD,24) TSMALL,TLARGE                                           Z  50
      READ (IRD,23) BDAT                                                    Z  51
      READ (IRD,22) RW,RDAT                                                 Z  52
      IBEGIN=ALOG10(TSMALL)                                                 Z  53
      IEND=ALOG10(TLARGE)+.99999                                            Z  54
      IF ((IBEGIN/2*2).LT.IBEGIN) IBEGIN=IBEGIN-1                           Z  55
      ISPAN=IEND-IBEGIN                                                     Z  56
      MLIMIT=(ISPAN+1)/2                                                    Z  57
C     COMPUTES INITIAL DELTA U (DU) = .001/SQRT(SMALLEST ALPHA)             Z  58
      DU=.001/SQRT(TDAT(1)*10.**IBEGIN)                                     Z  59
C     EXPONENT (JBEGIN) OF SMALLEST RW/B IS COMPUTED FROM EXPONENT          Z  60
C     (IBEGIN) OF SMALLEST ALPHA.                                           Z  61
      JBEGIN=-IBEGIN/2-2                                                    Z  62
      DO 1 I=1,13                                                           Z  63
      IF (BDAT(I).EQ.0.) GO TO 2                                            Z  64
    1 CONTINUE                                                              Z  65
      NB=13                                                                 Z  66
      GO TO 3                                                               Z  67
    2 NB=I-1                                                                Z  68
    3 CONTINUE                                                              Z  69
      DO 4 I=1,9                                                            Z  70
      IF (RDAT(I).EQ.0.) GO TO 5                                            Z  71
    4 RDAT(I)=RDAT(I)/RW                                                    Z  72
      NR=9                                                                  Z  73
      GO TO 6                                                               Z  74
    5 NR=I-1                                                                Z  75
    6 DO 21 M=1,MLIMIT                                                      Z  76
      NUM=8000                                                              Z  77
      START=DU/2.                                                           Z  78
      U=START                                                               Z  79
      DO 7 I=1,NUM                                                          Z  80
      U=U+DU                                                                Z  81
      CALL BESY(U,0,Y(I),IDUMY)                                             Z  82
    7 J(I)=J0(U)                                                            Z  83
      DO 19 IR=1,NR                                                         Z  84
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
                                                                    Z  85
      RHO=RDAT(IR)                                                  Z  86
      U=START                                                       Z  87
      DO 8 I=1,NUM                                                  Z  88
      U=U+DU                                                        Z  89
      CALL BESY(RHO*U,0,YOPU,IDUMY)                                 Z  90
      JOPU=J0(RHO*U)                                                Z  91
      JOU=J(I)                                                      Z  92
      YOU=Y(I)                                                      Z  93
    8 F(I)=(JOPU*YOU-YOPU*JOU)/(JOU*JOU+YOU*YOU)                    Z  94
      DO 19 IT=1,25                                                 Z  95
      INDEX=(IT-1)/6                                                Z  96
      IEXP=IBEGIN+INDEX                                             Z  97
      II=IT-INDEX*6                                                 Z  98
      TAU=TDAT(II)*10.**IEXP                                        Z  99
      T(IT)=TAU                                                     Z 100
      U=START                                                       Z 101
      NUMT=NUM                                                      Z 102
      DO 9 I=1,NUMT                                                 Z 103
      U=U+DU                                                        Z 104
      FTEST=F(I)                                                    Z 105
      IF (ABS(FTEST).LT.1.E-30) GO TO 10                            Z 106
      XTEST=-TAU*U*U                                                Z 107
      IF (XTEST+69.) 10,10,9                                        Z 108
    9 FT(I)=FTEST*EXP(XTEST)                                        Z 109
      GO TO 11                                                      Z 110
   10 NUMT=I-1                                                      Z 111
      FT(I)=0.                                                      Z 112
   11 DO 19 IB=1,13                                                 Z 113
      JNDEX=(IB-1)/NB                                               Z 114
      JEXP=JBEGIN+JNDEX                                             Z 115
      JJ=IB-JNDEX*NB                                                Z 116
      BETA=BDAT(JJ)*10.**JEXP                                       Z 117
      B(IB)=BETA                                                    Z 118
      U=START                                                       Z 119
      BSQ=BETA*BETA                                                 Z 120
      NUMB=NUMT                                                     Z 121
      DO 12 I=1,NUMB                                                Z 122
      U=U+DU                                                        Z 123
      FTEST=FT(I)                                                   Z 124
      IF (ABS(FTEST).LT.1.E-30) GO TO 13                            Z 125
   12 FB(I)=FTEST/(U+BSQ/U)                                         Z 126
      GO TO 14                                                      Z 127
   13 NUMB=I-1                                                      Z 128
      FB(I)=0.                                                      Z 129
   14 SUM1=0.                                                       Z 130
      SUM2=0.                                                       Z 131
      DO 15 I=1,NUMB,2                                              Z 132
      SUM1=SUM1+FB(I)                                               Z 133
   15 SUM2=SUM2+FB(I+1)                                             Z 134
      XINT=(SUM1+SUM2)*DU                                           Z 135
      CALL BESK(RHO*BETA,0,KOBP,IDUMY)                              Z 136
      CALL BESK(BETA,0,KOB,IDUMY)                                   Z 137
      RATIO=0.                                                      Z 138
      IF (KOBP.GT.0.) RATIO=KOBP/KOB                                Z 139
      XTEST=-TAU*BSQ                                                Z 140
      IF (XTEST+30.) 16,17,17                                       Z 141
   16 XPT=0.                                                        Z 142
      GO TO 18                                                      Z 143
   17 XPT=EXP(XTEST)                                                Z 144
   18 Z=RATIO+.6366198*XPT*XINT                                     Z 145
      IF ((Z.LT.0.).AND.(Z.GT.-5.E-5)) Z=0.E0                       Z 146
   19 ARRAY(IT,IR,IB)=Z
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
      DO 20 K=1,NR                                                      Z 147
      WRITE (IPT,25) RDAT(K),B                                          Z 148
      WRITE (IPT,26) (T(I),(ARRAY(I,K,L),L=1,13),I=1,25)                Z 149
   20 CONTINUE                                                          Z 150
C     EXPONENT OF SMALLEST R=/B DECREASED BY ONE EACH TIME THROUGH LOOP  Z 151
      JBEGIN=JBEGIN-1                                                   Z 152
C     EXPONENT OF SMALLEST ALPHA INCREASED BY TWO EACH TIME THROUGH LOOP Z 153
      IBEGIN=IBEGIN+2                                                   Z 154
C     DELTA U (DU) IS DIVIDED BY 10 EACH TIME THROUGH THE LOOP          Z 155
   21 DU=.1*DU                                                          Z 156
      STOP                                                              Z 157
C                                                                       Z 158
   22 FORMAT (10F8.2)                                                   Z 159
   23 FORMAT (13F5.0)                                                   Z 160
   24 FORMAT (2E10.5)                                                   Z 161
   25 FORMAT ('1','Z(ALPHA,R/RW,RW/B), R/RW=',F6.0/'0',9X,'I RW/B'/(' ', Z 162
     13X,'ALPHA I',13E9.2))                                             Z 163
   26 FORMAT (' ',E10.3,13F9.3)                                         Z 164
      END                                                               Z 165-
      REAL FUNCTION J0*4(X)                                            J0   1
C     ********************************************************************** J0   2
C                                                                            J0   3
C     FUNCTION J0                                                            J0   4
C                                                                            J0   5
C     PURPOSE                                                                J0   6
C        TO COMPUTE THE ZERO ORDER J BESSEL FUNCTION FOR A GIVEN             J0   7
C        ARGUMENT.                                                           J0   8
C     USAGE                                                                  J0   9
C        J0(X)                                                               J0  10
C     DESCRIPTION OF PARAMETER                                               J0  11
C        X - REAL*4, ARGUMENT OF J0 BESSEL FUNCTION DESIRED.                 J0  12
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                          J0  13
C        NONE.                                                               J0  14
C     METHOD                                                                 J0  15
C        POLYNOMIAL APPROXIMATION FOR X<4 AND ASYMPTOTIC SERIES FOR          J0  16
C        X GE 4. THE POLYNOMIAL APPROXIMATION IS THE FIRST 10 TERMS OF       J0  17
C        THE POWER SERIES FOR J0(X) (MILLER, K.S., 1957,                     J0  18
C        ENGINEERING MATHEMATICS: RINEHART AND CO., INC., NEW YORK,          J0  19
C        P. 120). THE ASYMPTOTIC EXPANSION OF J0(X) IS GIVEN ON P. 82        J0  20
C        OF BOWMAN, FRANK, 1958, INTRODUCTION TO BESSEL FUNCTIONS:           J0  21
C        DOVER PUBLICATIONS INC., NEW YORK. THE TERMS P ('A*P0') AND         J0  22
C        Q ('B*Q0') OF THE ASYMPTOTIC EXPANSION ARE COMPUTED BY AN           J0  23
C        ALGORITHM FROM IBM SUBROUTINE BESY.                                 J0  24
C                                                                            J0  25
C     ********************************************************************** J0  26
      IF (X-4.) 1,3,3                                                        J0  27
C     COMPUTE J0 BY FIRST 10 TERMS OF POWER SERIES                           J0  28
    1 A=-X*X/4.                                                              J0  29
      B=1.                                                                   J0  30
      DO 2 I=1,10                                                            J0  31
      C=11.-I                                                                J0  32
    2 B=1.+B*(A/(C*C))                                                       J0  33
      J0=B                                                                   J0  34
      GO TO 4                                                                J0  35
C     COMPUTE J0 BY ASYMPTOTIC SERIES                                        J0  36
    3 T1=4./X                                                                J0  37
      T2=T1*T1                                                               J0  38
      P0=((((-.0000037043*T2+.0000173565)*T2-.0000487613)*T2+.00017343)*     J0  39
     1T2-.001753062)*T2+.3989423                                             J0  40
      Q0=((((.00000032312*T2-.0000142078)*T2+.0000342468)*T2-.0000869791)    J0  41
     1*T2+.00004564324)*T2-.01246694                                         J0  42
      A=2.0/SQRT(X)                                                          J0  43
```

best copy available

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
      B=A*T1                                                              J0  44
      C=X-.7853982                                                        J0  45
      J0=A*P0*COS(C)-B*Q0*SIN(C)                                          J0  46
    4 RETURN                                                              J0  47
      END                                                                 J0  48
      SUBROUTINE BESY(X,N,BY,IER)                                         BESY 410
                                                                          BESY  10
C     ******************************************************************BESY  20
C                                                                         BESY  30
C        SUBROUTINE BESY                                                  BESY  40
C                                                                         BESY  50
C        PURPOSE                                                          BESY  60
C           COMPUTE THE Y BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDER  BESY  70
C                                                                         BESY  80
C        USAGE                                                            BESY  90
C           CALL BESY(X,N,BY,IER)                                         BESY 100
C                                                                         BESY 110
C        DESCRIPTION OF PARAMETERS                                        BESY 120
C           X  -THE ARGUMENT OF THE Y BESSEL FUNCTION DESIRED             BESY 130
C           N  -THE ORDER OF THE Y BESSEL FUNCTION DESIRED                BESY 140
C           BY -THE RESULTANT Y BESSEL FUNCTION                           BESY 150
C           IER-RESULTANT ERROR CODE WHERE                                BESY 160
C              IER=0  NO ERROR                                            BESY 170
C              IER=1  N IS NEGATIVE                                       BESY 180
C              IER=2  X IS NEGATIVE OR ZERO                               BESY 190
C              IER=3  BY HAS EXCEEDED MAGNITUDE OF 10**70                 BESY 200
C                                                                         BESY 210
C        REMARKS                                                          BESY 220
C           VERY SMALL VALUES OF X MAY CAUSE THE RANGE OF THE LIBRARY     BESY 230
C           FUNCTION ALOG TO BE EXCEEDED                                  BESY 240
C           X MUST BE GREATER THAN ZERO                                   BESY 250
C           N MUST BE GREATER THAN OR EQUAL TO ZERO                       BESY 260
C                                                                         BESY 270
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                    BESY 280
C           NONE                                                          BESY 290
C                                                                         BESY 300
C                                                                         BESY 310
C        METHOD                                                           BESY 320
C           RECURRENCE RELATION AND POLYNOMIAL APPROXIMATION TECHNIQUE    BESY 330
C           AS DESCRIBED BY A,J,M,HITCHCOCK,'POLYNOMIAL APPROXIMATIONS    BESY 340
C           TO BESSEL FUNCTIONS OF ORDER ZERO AND ONE AND TO RELATED      BESY 350
C           FUNCTIONS', M,T,A,C., V,11,1957,PP,86-88, AND G,N, WATSON,    BESY 360
C           'A TREATISE ON THE THEORY OF BESSEL FUNCTIONS', CAMBRIDGE     BESY 370
C           UNIVERSITY PRESS, 1958, P, 62                                 BESY 380
C     ******************************************************************BESY 390
                                                                          BESY 400
                                                                          BESY 410
C     CHECK FOR ERRORS IN N AND X                                         BESY 420
C                                                                         BESY 430
      IF(N)180,10,10                                                      BESY 440
   10 IER=0                                                               BESY 450
      IF(X)190,190,20                                                     BESY 460
C                                                                         BESY 470
C     BRANCH IF X LESS THAN OR EQUAL 4                                    BESY 480
C                                                                         BESY 490
   20 IF(X-4.0)40,40,30                                                   BESY 500
C                                                                         BESY 510
C     COMPUTE Y0 AND Y1 FOR X GREATER THAN 4                              BESY 520
C                                                                         BESY 530
   30 T1=4.0/X                                                            BESY 540
      T2=T1*T1                                                            BESY 550
      P0=((((-.0000037043*T2+.0000173565)*T2-.0000487613)*T2              BESY 560
                                                                          BESY 570
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
     1  +,00017343)*T2=,001753062)*T2+,3989423                                BESY 580
      Q0=((((,0000032312*T2=,0000142078)*T2+,0000342468)*T2                   BESY 590
     1  =,0000869791)*T2+,0004564324)*T2=,01246694                            BESY 600
      P1=((((,0000042414*T2=,0000200920)*T2+,0000580759)*T2                   BESY 610
     1  =,000223203)*T2+,002921826)*T2+,3989423                               BESY 620
      Q1=((((=,0000036594*T2+,00001622)*T2=,0000398708)*T2                    BESY 630
     1  +,00010647#1)*T2=,0006390400)*T2+,03740084                            BESY 640
      A=2,0/SQRT(X)                                                           BESY 650
      B=A*T                                                                   BESY 660
      C=X=,7853982                                                            BESY 670
      Y0=A*P0*SIN(C)+B*Q0*COS(C)                                              BESY 680
      Y1=-A*P1*COS(C)+B*Q1*SIN(C)                                             BESY 690
      GO TO 90                                                                BESY 700
C                                                                             BESY 710
C       COMPUTE Y0 AND Y1 FOR X LESS THAN OR EQUAL TO 4                       BESY 720
C                                                                             BESY 730
   40 XX=X/2,                                                                 BESY 740
      X2=XX*XX                                                                BESY 750
      T=ALOG(XX)+,5772157                                                     BESY 760
      SUM=0,                                                                  BESY 770
      TERM=T                                                                  BESY 780
      Y0=T                                                                    BESY 790
      DO 70 L=1,15                                                            BESY 800
      IF(L=1)50,60,50                                                         BESY 810
   50 SUM=SUM+1,/FLOAT(L=1)                                                   BESY 820
   60 FL=L                                                                    BESY 830
      TS=T=SUM                                                                BESY 840
      IF(ABS(TERM),LE,1,E=40) TERM=0,                                         BESY 841
      TERM=(TERM*(=X2)/FL**2)*(1,=1,/(FL*TS))                                 BESY 850
   70 Y0=Y0+TERM                                                              BESY 860
      TERM = XX*(T=,5)                                                        BESY 870
      SUM=0,                                                                  BESY 880
      Y1=TERM                                                                 BESY 890
      DO 80 L=2,16                                                            BESY 900
      SUM=SUM+1,/FLOAT(L=1)                                                   BESY 910
      FL=L                                                                    BESY 920
      FL1=FL=1,                                                               BESY 930
      TS=T=SUM                                                                BESY 940
      IF(ABS(TERM),LE,1,E=40) TERM=0,                                         BESY 941
      TERM=(TERM*(=X2)/(FL1*FL))*((TS=,5/FL)/(TS+,5/FL1))                     BESY 950
   80 Y1=Y1+TERM                                                              BESY 960
      PI2=,6366198                                                            BESY 970
      Y0=PI2*Y0                                                               BESY 980
      Y1=-PI2/X+PI2*Y1                                                        BESY 990
C                                                                             BESY1000
C     CHECK IF ONLY Y0 OR Y1 IS DESIRED                                       BESY1010
C                                                                             BESY1020
   90 IF(N=1)100,100,130                                                      BESY1030
C                                                                             BESY1040
C     RETURN EITHER Y0 OR Y1 AS REQUIRED                                      BESY1050
C                                                                             BESY1060
  100 IF(N)110,120,110                                                        BESY1070
  110 BY=Y1                                                                   BESY1080
      GO TO 170                                                               BESY1090
  120 BY=Y0                                                                   BESY1100
      GO TO 170                                                               BESY1110
C                                                                             BESY1120
C    PERFORM RECURRENCE OPERATIONS TO FIND YN(X)                              BESY1130
C                                                                             BESY1140
  130 YA=Y0                                                                   BESY1150
      YB=Y1                                                                   BESY1160
      K=1                                                                     BESY1170
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
  140 T=FLOAT(2*K)/X                                                    BESY1180
      YC=T*YB-YA                                                        BESY1190
      IF(ABS(YC)-1.0E70)145,145,141                                     BESY1200
  141 IER=3                                                             BESY1210
      RETURN                                                            BESY1220
  145 K=K+1                                                             BESY1230
      IF(K-N)150,160,150                                                BESY1240
  150 YA=YB                                                             BESY1250
      YB=YC                                                             BESY1260
      GO TO 140                                                         BESY1270
  160 BY=YC                                                             BESY1280
  170 RETURN                                                            BESY1290
  180 IER=1                                                             BESY1300
      RETURN                                                            BESY1310
  190 IER=2                                                             BESY1320
      RETURN                                                            BESY1330
      END                                                               BESY134*
      SUBROUTINE BESK(X,N,BK,IER)                                       BESK  410
C                                                                       BESK   10
C     ..................................................................BESK   20
C                                                                       BESK   30
C     SUBROUTINE BESK                                                   BESK   40
C                                                                       BESK   50
C        COMPUTE THE K BESSEL FUNCTION FOR A GIVEN ARGUMENT AND ORDER   BESK   60
C                                                                       BESK   70
C     USAGE                                                             BESK   80
C        CALL BESK(X,N,BK,IER)                                          BESK   90
C                                                                       BESK  100
C     DESCRIPTION OF PARAMETERS                                         BESK  110
C        X  -THE ARGUMENT OF THE K BESSEL FUNCTION DESIRED              BESK  120
C        N  -THE ORDER OF THE K BESSEL FUNCTION DESIRED                 BESK  130
C        BK -THE RESULTANT K BESSEL FUNCTION                            BESK  140
C        IER-RESULTANT ERROR CODE WHERE                                 BESK  150
C           IER=0  NO ERROR                                             BESK  160
C           IER=1  N IS NEGATIVE                                        BESK  170
C           IER=2  X IS ZERO OR NEGATIVE                                BESK  180
C           IER=3  X .GT. 170, MACHINE RANGE EXCEEDED                   BESK  190
C           IER=4  BK .GT. 10**70                                       BESK  200
C                                                                       BESK  210
C     REMARKS                                                           BESK  220
C        N MUST BE GREATER THAN OR EQUAL TO ZERO                        BESK  230
C                                                                       BESK  240
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                     BESK  250
C        NONE                                                           BESK  260
C                                                                       BESK  270
C                                                                       BESK  280
C     METHOD                                                            BESK  290
C        COMPUTES ZERO ORDER AND FIRST ORDER BESSEL FUNCTIONS USING     BESK  300
C        SERIES APPROXIMATIONS AND THEN COMPUTES N TH ORDER FUNCTION    BESK  310
C        USING RECURRENCE RELATION,                                     BESK  320
C        RECURRENCE RELATION AND POLYNOMIAL APPROXIMATION TECHNIQUE     BESK  330
C        AS DESCRIBED BY A,J,H,HITCHCOCK,'POLYNOMIAL APPROXIMATIONS     BESK  340
C        TO BESSEL FUNCTIONS OF ORDER ZERO AND ONE AND TO RELATED       BESK  350
C        FUNCTIONS', M,T,A,C., V,11,1957,PP,86-88, AND G,N, WATSON,     BESK  360
C        'A TREATISE ON THE THEORY OF BESSEL FUNCTIONS', CAMBRIDGE      BESK  370
C        UNIVERSITY PRESS, 1958, P, 62                                  BESK  380
C                                                                       BESK  390
C     ..................................................................BESK  400
C                                                                       BESK  420
      DIMENSION T(12)                                                   BESK  430
      BK=.0                                                             BESK  440
      IF(N)10,11,11                                                     BESK  450
   10 IER=1
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
      RETURN                                                          BESK 460
   11 IF(X)12,12,20                                                   BESK 470
   12 IER=2                                                           BESK 480
      RETURN                                                          BESK 490
   20 IF(X-170.0)22,22,21                                             BESK 500
   21 IER=3                                                           BESK 510
      RETURN                                                          BESK 520
   22 IER=0                                                           BESK 530
      IF(X-1.)36,36,25                                                BESK 540
   25 A=EXP(-X)                                                       BESK 550
      B=1./X                                                          BESK 560
      C=SQRT(B)                                                       BESK 570
      T(1)=B                                                          BESK 580
      DO 26 L=2,12                                                    BESK 590
   26 T(L)=T(L-1)*B                                                   BESK 600
      IF(N-1)27,29,27                                                 BESK 610
C                                                                     BESK 620
C     COMPUTE K0 USING POLYNOMIAL APPROXIMATION                       BESK 630
C                                                                     BESK 640
   27 G0=A*(1.2533141-.1566642*T(1)+.08811128*T(2)-.09139095*T(3)     BESK 650
     2+.1344596*T(4)-.2299850*T(5)+.3792410*T(6)-.5247277*T(7)        BESK 660
     3+.5575368*T(8)-.4262633*T(9)+.2184518*T(10)-.06680977*T(11)     BESK 670
     4+.009189383*T(12))*C                                            BESK 680
      IF(N)20,28,29                                                   BESK 690
   28 BK=G0                                                           BESK 700
      RETURN                                                          BESK 710
C                                                                     BESK 720
C     COMPUTE K1 USING POLYNOMIAL APPROXIMATION                       BESK 730
C                                                                     BESK 740
   29 G1=A*(1.2533141+.4699927*T(1)-.1468583*T(2)+.1280427*T(3)       BESK 750
     2-.1736432*T(4)+.2847618*T(5)-.4594342*T(6)+.6283381*T(7)        BESK 760
     3-.6632295*T(8)+.5050239*T(9)-.2581304*T(10)+.07880001*T(11)     BESK 770
     4-.01082418*T(12))*C                                             BESK 780
      IF(N-1)20,30,31                                                 BESK 790
   30 BK=G1                                                           BESK 800
      RETURN                                                          BESK 810
C                                                                     BESK 820
C     FROM K0,K1 COMPUTE KN USING RECURRENCE RELATION                 BESK 830
C                                                                     BESK 840
   31 DO 35 J=2,N                                                     BESK 850
      GJ=2.*(FLOAT(J)-1.)*G1/X+G0                                     BESK 860
      IF(GJ-1.0E70)33,33,32                                           BESK 870
   32 IER=4                                                           BESK 880
      GO TO 34                                                        BESK 890
   33 G0=G1                                                           BESK 900
   35 G1=GJ                                                           BESK 910
   34 BK=GJ                                                           BESK 920
      RETURN                                                          BESK 930
   36 B=X/2.                                                          BESK 940
      A=.5772157+ALOG(B)                                              BESK 950
      C=B*B                                                           BESK 960
      IF(N-1)37,43,37                                                 BESK 970
C                                                                     BESK 980
C     COMPUTE K0 USING SERIES EXPANSION                               BESK 990
C                                                                     BESK1000
   37 G0=-A                                                           BESK1010
      X2J=1.                                                          BESK1020
      FACT=1.                                                         BESK1030
      HJ=.0                                                           BESK1040
      DO 40 J=1,6                                                     BESK1050
      RJ=1./FLOAT(J)                                                  BESK1060
      IF(X2J.LT.1.E-40) X2J=0.                                        BESK1061
C     PREVIOUS STATEMENT ADDED TO IBM SUBROUTINE TO CORRECT UNDERFLOW BESK1062
```

TABLE 7.2.—*Listing of program for constant drawdown in a well in an infinite leaky aquifer*—Continued

```
C     PROBLEM ON WATFOR COMPILER                                        BESK1063
      X2J=X2J*C                                                         BESK1070
      FACT=FACT*RJ*RJ                                                   BESK1080
      HJ=HJ+RJ                                                          BESK1090
   40 G0=G0+X2J*FACT*(HJ-A)                                             BESK1100
      IF(N)43,42,43                                                     BESK1110
   42 BK=G0                                                             BESK1120
      RETURN                                                            BESK1130
C                                                                       BESK1140
C     COMPUTE K1 USING SERIES EXPANSION                                 BESK1150
C                                                                       BESK1160
   43 X2J=B                                                             BESK1170
      FACT=1.                                                           BESK1180
      HJ=1.                                                             BESK1190
      G1=1./X+X2J*(.5+A-HJ)                                             BESK1200
      DO 50 J=2,8                                                       BESK1210
      X2J=X2J*C                                                         BESK1220
      RJ=1./FLOAT(J)                                                    BESK1230
      FACT=FACT*RJ*RJ                                                   BESK1240
      HJ=HJ+RJ                                                          BESK1250
   50 G1=G1+X2J*FACT*(.5+(A-HJ)*FLOAT(J))                               BESK1260
      IF(N-1)51,52,51                                                   BESK1270
   52 BK=G1                                                             BESK1280
      RETURN                                                            BESK1290
      END                                                               BESK130-
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*

```
C*************************************************************************FAR   1
C                                                                         FAR   2
C     PURPOSE                                                             FAR   3
C       COMPUTES FUNCTION VALUES OF F(U,ALPHA,RHO) FOR RHO > 1 -          FAR   4
C       PAPADOPULOS,I.S. AND COOPER,H.H.,JR., 1967, DRAWDOWN IN           FAR   5
C       A WELL OF LARGE DIAMETER: WATER RESOURCES RESEARCH, V. 3,         FAR   6
C       NO. 1, P. 241-244.                                                FAR   7
C       PROGRAM BY I.S.PAPADOPULOS.                                       FAR   8
C     INPUT DATA - ONE OR MORE GROUPS, EACH GROUP CODED AS FOLLOWS        FAR   9
C       1 CARD - FORMAT(2E10.5)                                           FAR  10
C            ALPHA  -      RW**2*S/RC**2 - RADIUS OF WELL (SCREEN       FAR  11
C                 OR OPEN BORE IN AQUIFER) SQUARED * STORAGE             FAR  12
C                 COEFFICIENT / RADIUS OF CASING (OVER INTERVAL OF       FAR  13
C                 WATER LEVEL CHANGE) SQUARED.                          FAR  14
C            RHO - R/RW - DISTANCE FROM PUMPED WELL / RADIUS OF          FAR  15
C                 WELL (SCREEN OR OPEN BORE IN AQUIFER), MUST BE        FAR  16
C                 GREATER THAN ONE.                                     FAR  17
C       1 CARD - FORMAT(16E5.0)                                           FAR  18
C            U- 16 VALUES OF U - R**2*S/(4*T*TIME) - DISTANCE FROM       FAR  19
C                 PUMPED WELL SQUARED * STORAGE COEFFICIENT /           FAR  20
C                 4 * TRANSMISSIVITY * TIME. IF LESS THAN 16 DESIRED,   FAR  21
C                 BLANK OR ZERO VALUES MAY BE CODED FOR THE REST.       FAR  22
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                       FAR  23
C       PEAK,SIMP,APEKE,EXBSL1,JY0,JY1,ROOTS - MUST BE IN DECK.           FAR  24
C                                                                         FAR  25
C*************************************************************************FAR  26
      DIMENSION V(40,40),U(16)                                            FAR  27
      COMMON XPK,YPK                                                      FAR  28
      COMMON/PBLK/A,B,RHO                                                 FAR  29
      EXTERNAL EXBSL1                                                     FAR  30
    1 READ (5,16,END=15) ALPHA,RHO                                        FAR  31
      IF (ALPHA) 15,15,2                                                  FAR  32
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*—Continued

```
      2 WRITE (6,17) ALPHA,RHO                                                FAR  33
        WRITE (6,18)                                                          FAR  34
      3 READ (5,19) U                                                         FAR  35
        DO 14 II=1,16                                                         FAR  36
        IF (U(II)) 1,1,4                                                      FAR  37
      4 A=ALPHA+ALPHA                                                         FAR  38
        B=0.25/U(II)                                                          FAR  39
        CALL APEKE(EXBSL1)                                                    FAR  40
        CALL PEAK(EXBSL1)                                                     FAR  41
        IF (XPK-1.0E-8) 5,6,6                                                 FAR  42
      5 WRITE (6,20) XPK,U                                                    FAR  43
        GO TO 3                                                               FAR  44
      6 IF (XPK-3.0) 8,7,7                                                    FAR  45
      7 WRITE (6,21) XPK,U                                                    FAR  46
        GO TO 3                                                               FAR  47
      8 EPS=0.000001                                                          FAR  48
        HBAR=0.007*XPK                                                        FAR  49
        CALL SIMPS(0.0,XPK,EPS,HBAR,SUM,DEL,EXBSL1)                           FAR  50
        XM1=((3.14159265*7.0)/(8.0*(RHO-1.)))+1.E-6)*RHO/2.                   FAR  51
        DX1=XM1-(1.0E-6)*RHO                                                  FAR  52
        DXN=(2.0*3.14159265*RHO)/(5.*(RHO-1.))                                FAR  53
        DL=3.14159265*RHO/(RHO-1.)                                            FAR  54
        CALL ROOTS(XM1,DX1,RT1,EXBSL1)                                        FAR  55
        HBAR=0.007*(RT1-XPK)                                                  FAR  56
        CALL SIMPS(XPK,RT1,EPS,HBAR,TRM1,ERR1,EXBSL1)                         FAR  57
        SUM=SUM+TRM1                                                          FAR  58
        DEL=DEL+ERR1                                                          FAR  59
        X1=RT1                                                                FAR  60
        I=1                                                                   FAR  61
      9 XM=X1+DL                                                              FAR  62
        CALL ROOTS(XM,DXN,X2,EXBSL1)                                          FAR  63
        HBAR=0.007*(X2-X1)                                                    FAR  64
        CALL SIMPS(X1,X2,EPS,HBAR,TRM,ERR,EXBSL1)                             FAR  65
        V(I,I)=ABS(TRM)                                                       FAR  66
        DEL=DEL+ERR                                                           FAR  67
        I=I+1                                                                 FAR  68
        IF (I-40) 10,10,11                                                    FAR  69
     10 X1=X2                                                                 FAR  70
        GO TO 9                                                               FAR  71
     11 EST=0.0                                                               FAR  72
        DO 12 K=2,40                                                          FAR  73
        M=41-K                                                                FAR  74
        DO 12 J=1,M                                                           FAR  75
     12 V(K,J)=V(K-1,J+1)-V(K-1,J)                                            FAR  76
        DO 13 N=1,40                                                          FAR  77
        L=N-1                                                                 FAR  78
        DELV=(-0.5)**L*V(N,1)                                                 FAR  79
     13 EST=EST+(0.5)*DELV                                                    FAR  80
        SUM=SUM-EST                                                           FAR  81
        PUAR=4.0*A*RHO*SUM/3.14159265                                         FAR  82
        WRITE (6,22) U(II),SUM,DEL,PUAR                                       FAR  83
     14 CONTINUE                                                              FAR  84
        GO TO 1                                                               FAR  85
     15 STOP                                                                  FAR  86
C                                                                             FAR  87
     16 FORMAT (2E10.5)                                                       FAR  88
     17 FORMAT ('1','F(U,ALPHA,RHO) FOR ALPHA=',1PE13.5,', RHO=',1E13.5)      FAR  89
     18 FORMAT (1H0,12X,1HU,16X,8HINTEGRAL,9X,14HINTEGRAL ERROR,6X,14HF(U,    FAR  90
       1ALPHA,RHO)/1H )                                                       FAR  91
     19 FORMAT (16E5.0)                                                       FAR  92
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*—Continued

```
   20 FORMAT (5H XPK=,E15.8,3X,16HTOO SMALL FOR U=,E10.3)                    FAR  93
   21 FORMAT (5H XPK=,E15.8,3X,16HTOO LARGE FOR U=,E10.3)                    FAR  94
   22 FORMAT (1H ,1P4E20.8)                                                  FAR  95
      END                                                                    FAR  96
      FUNCTION EXBSL1(X)                                                     EB1   1
C*************************************************************************EB1   2
C                                                                           EB1   3
C                                                                           EB1   4
C     PURPOSE                                                               EB1   5
C        COMPUTES VALUES OF THE INTEGRAND FOR F(U,ALPHA,RHO)                EB1   6
C     DESCRIPTION OF PARAMETER                                              EB1   7
C        X= REAL = ARGUMENT OF INTEGRAND                                    EB1   8
C                                                                           EB1   9
C*************************************************************************EB1  10
      COMMON/PBLK/A,B,R                                                     EB1  11
      IF (X) 1,1,2                                                          EB1  12
    1 EXBSL1=0.                                                             EB1  13
      GO TO 8                                                               EB1  14
    2 W=X/R                                                                 EB1  15
      IF (W-1.0E7) 3,4,4                                                    EB1  16
    3 FNU=A*COS(W*(R-1.0))-W*SIN(W*(R-1.0))                                 EB1  17
      DE=(W*W*SQRT(R))*(W*W+A*A)                                            EB1  18
      EXBSL1=FNU/DE                                                         EB1  19
      GO TO 8                                                               EB1  20
    4 Y=B*X*X                                                               EB1  21
      IF (Y-0.01) 5,5,6                                                     EB1  22
    5 EXPO=Y*(1.0-Y*(0.5-Y*((1.0/6.0)-Y*(1.0/24.0))))                       EB1  23
      GO TO 7                                                               EB1  24
    6 EXPO=1.0-EXP(-Y)                                                      EB1  25
    7 CALL JY0(W,WJ0,WY0)                                                   EB1  26
      CALL JY1(W,WJ1,WY1)                                                   EB1  27
      AW=W*WY0-A*WY1                                                        EB1  28
      BW=W*WJ0-A*WJ1                                                        EB1  29
      CALL JY0(X,BJ0,BY0)                                                   EB1  30
      FNUM=EXPO*(AW*BJ0-BW*BY0)                                             EB1  31
      DEN=X*X*(AW*AW+BW*BW)                                                 EB1  32
      EXBSL1=FNUM/DEN                                                       EB1  33
    8 RETURN                                                                EB1  34
      END                                                                   ROO   1
      SUBROUTINE ROOTS(XM,DX,ROOT,F)                                        ROO   2
C*************************************************************************ROO   3
C                                                                           ROO   4
C     PURPOSE                                                               ROO   5
C        SEARCHES FOR ROOT OF F IN THE INTERVAL XM-DX TO XM+DX.             ROO   6
C     DESCRIPTION OF PARAMETERS - ALL REAL                                  ROO   7
C        XM - CENTER OF INTERVAL SEARCHED.                                  ROO   8
C        DX - HALF WIDTH OF INTERVAL SEARCHED.                              ROO   9
C        ROOT - RETURNED ROOT LOCATION.                                     ROO  10
C        F - FUNCTION REFERENCE.                                            ROO  11
C                                                                           ROO  12
C*************************************************************************ROO  13
      XL=XM-DX                                                              ROO  14
      XR=XM+DX                                                              ROO  15
      YL=F(XL)                                                              ROO  16
      YR=F(XR)                                                              ROO  17
      EP=0.000001*ABS(YL)                                                   ROO  18
      DO 9 I=1,200                                                          ROO  19
      YM=F(XM)                                                              ROO  20
      UP=ABS(YM)                                                            ROO  21
      IF (UP.LT.EP.AND.UP.LT.1.0D-7) GO TO 1                                ROO  22
      IF (YM) 2,1,2
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*—Continued

```
    1 ROOT=XM                                                              ROO  23
      GO TO 10                                                             ROO  24
    2 IF (YM*YL) 7,3,4                                                     ROO  25
    3 ROOT=XL                                                              ROO  26
      GO TO 10                                                             ROO  27
    4 IF (YM*YR) 8,5,6                                                     ROO  28
    5 ROOT=XR                                                              ROO  29
      GO TO 10                                                             ROO  30
    6 WRITE (6,11) XL,XR                                                   ROO  31
      STOP                                                                 ROO  32
    7 XR=XM                                                                ROO  33
      YR=YM                                                                ROO  34
      GO TO 9                                                              ROO  35
    8 XL=XM                                                                ROO  36
      YL=YM                                                                ROO  37
    9 XM=(XL+XR)/2.0                                                       ROO  38
      ROOT=XM                                                              ROO  39
   10 RETURN                                                               ROO  40
C                                                                          ROO  41
   11 FORMAT (1H ,10X,27HNO ROOT IN INTERVAL XM-DX =,1PE20.8,5X,11HAND XROO  42
     1M+DX =,1PE20.8/)                                                     ROO  43
      END                                                                  ROO  44*
      SUBROUTINE APEKE(EXBSL)                                              APE   1
C*************************************************************************APE   2
C                                                                          APE   3
C     PURPOSE                                                              APE   4
C        GETS FIRST APPROXIMATION TO PEAK POSITION                         APE   5
C                                                                          APE   6
C*************************************************************************APE   7
      COMMON XPK,YPK                                                       APE   8
      XPK=0.0                                                              APE   9
      YPK=0.0                                                              APE  10
      DO 2 I=1,17                                                          APE  11
      X=10.0**(I-9)                                                        APE  12
      Y=EXBSL(X)                                                           APE  13
      IF (Y-YPK) 3,3,1                                                     APE  14
    1 XPK=X                                                                APE  15
      YPK=Y                                                                APE  16
    2 CONTINUE                                                             APE  17
    3 RETURN                                                               APE  18
      END                                                                  APE  19*
      SUBROUTINE PEAK(EXBSL)                                               PEA   1
C*************************************************************************PEA   2
C                                                                          PEA   3
C     PURPOSE                                                              PEA   4
C        ATTEMPTS TO FIND POSITION OF MAXIMUM FOR INTEGRAND                PEA   5
C                                                                          PEA   6
C*************************************************************************PEA   7
      COMMON XPK,YPK                                                       PEA   8
      YPK=EXBSL(XPK)                                                       PEA   9
      DO 13 L=1,200                                                        PEA  10
      DX=0.01*XPK                                                          PEA  11
      XL=XPK-DX                                                            PEA  12
      YL=EXBSL(XL)                                                         PEA  13
      XR=XPK+DX                                                            PEA  14
      YR=EXBSL(XR)                                                         PEA  15
      DEN=YR+YL-YPK-YPK                                                    PEA  16
      IF (DEN) 1,9,1                                                       PEA  17
    1 X=XPK-0.5*(YR-YL)*DX/DEN                                             PEA  18
    2 IF (X) 3,4,4                                                         PEA  19
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*—Continued

```
   3 X=0.0                                                                  PEA  20
   4 Y=EXBSL(X)                                                             PEA  21
     IF (YR-Y) 6,6,5                                                        PEA  22
   5 Y=YR                                                                   PEA  23
     X=XR                                                                   PEA  24
   6 IF (YL-Y) 8,8,7                                                        PEA  25
   7 Y=YL                                                                   PEA  26
     X=XL                                                                   PEA  27
   8 IF (Y-YPK) 14,14,12                                                    PEA  28
   9 IF (YR-YPK) 11,10,10                                                   PEA  29
  10 X=XPK+DX+DX                                                            PEA  30
     GO TO 2                                                                PEA  31
  11 X=XPK-DX-DX                                                            PEA  32
     GO TO 2                                                                PEA  33
  12 YPK=Y                                                                  PEA  34
     XPK=X                                                                  PEA  35
  13 CONTINUE                                                               PEA  36
  14 RETURN                                                                 PEA  37
     END                                                                    PEA  38
     SUBROUTINE SIMPS(Q,R,EPS,HBAR,AREA,DEL,F)                              SIM   1
C*****************************************************************************SIM   2
C                                                                           SIM   3
C    PURPOSE                                                                SIM   4
C      TO DETERMINE THE INTEGRAL OF A FUNCTION, F, FROM Q TO R,             SIM   5
C      USING SIMPSON'S RULE.                                                SIM   6
C    DESCRIPTION OF PARAMETERS                                              SIM   7
C      ALL REAL                                                             SIM   8
C      Q - LOWER LIMIT OF INTEGRAL                                          SIM   9
C      R - UPPER LIMIT OF INTEGRAL                                          SIM  10
C      EPS - DESIRED ACCURACY                                               SIM  11
C      HBAR - MINIMUM DIVISION OF THE INTERVAL                              SIM  12
C      AREA - COMPUTED VALUE OF INTEGRAL BETWEEN Q AND R                    SIM  13
C      DEL - COMPUTED ESTIMATE OF ERROR                                     SIM  14
C      F- THE INTEGRAND (FUNCTION REFERENCE)                                SIM  15
C    METHOD                                                                 SIM  16
C      USES SIMPSON'S RULE TO COMPUTE A SUM APPROXIMATING THE INTEGRAL      SIM  17
C      USES INITIAL H=(R-Q)/2, COMPUTES A SEQUENCE OF SUMS BY HALVING       SIM  18
C      H EACH TIME, COMPUTES ESTIMATE OF ERROR (DEL) AS (PREVIOUS           SIM  19
C      SUM - CURRENT SUM)/15, COMPUTATION STOPS WHEN 1) H<HBAR,             SIM  20
C      2) ABS(DEL)<ABS(EPS*CURRENT SUM), IF HBAR IS LE 0,                   SIM  21
C      THEN HBAR=.007*(R-Q).                                                SIM  22
C                                                                           SIM  23
C*****************************************************************************SIM  24
                                                                            SIM  25
     H=R-Q                                                                  SIM  26
     IF (H) 1,1,2                                                           SIM  27
   1 AREA=0.0                                                               SIM  28
     DEL=0.0                                                                SIM  29
     GO TO 10                                                               SIM  30
C R MUST BE GREATER THAN Q                                                  SIM  31
   2 SP=1.0E35                                                              SIM  32
     S3=0.0                                                                 SIM  33
     S1=F(Q)+F(R)                                                           SIM  34
     IF (HBAR) 3,3,4                                                        SIM  35
   3 HBAR=0.007*H                                                           SIM  36
   4 S2=0.0                                                                 SIM  37
     X=Q+0.5*H                                                              SIM  38
   5 S2=S2+4.0*F(X)                                                         SIM  39
     X=X+H                                                                  SIM  40
     IF (X-R) 5,5,6                                                         SIM  41
   6 SC=(S1+S2+S3)*H*0.16666667
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*—Continued

```
      DEL=0,066666667*(SP-SC)                                                  SIM  42
      IF (ABS(DEL)-ABS(EPS*SC)) 7,8,8                                          SIM  43
    7 AREA=SC-DEL                                                              SIM  44
      GO TO 10                                                                 SIM  45
    8 S3=S3+0,5*S2                                                             SIM  46
      H=0,5*H                                                                  SIM  47
      IF (H-HBAR) 7,9,9                                                        SIM  48
    9 SP=SC                                                                    SIM  49
      GO TO 4                                                                  SIM  50
   10 RETURN                                                                   SIM  51
      END                                                                      SIM  52-
      SUBROUTINE JY0(X,J0,Y0)                                                  JY0   1
C*************************************************************************** JY0   2
C                                                                              JY0   3
C     PURPOSE                                                                  JY0   4
C        COMPUTES BESSEL FUNCTIONS OF THE FIRST AND SECOND KIND,               JY0   5
C        ZERO ORDER, FOR POSITIVE ARGUMENTS,                                   JY0   6
C        SEE NBS AMS 55, P. 369-370.                                           JY0   7
C     DESCRIPTION OF PARAMETERS - ALL REAL                                     JY0   8
C        X- ARGUMENT, MUST BE >0                                               JY0   9
C        J0 - RETURNED FUNCTION VALUE, J0(X)                                   JY0  10
C        Y0 - RETURNED FUNCTION VALUE, Y0(X)                                   JY0  11
C                                                                              JY0  12
C*************************************************************************** JY0  13
      REAL J0                                                                  JY0  14
      IF (X-3,0) 1,2,3                                                         JY0  15
    1 IF (X) 4,4,2                                                             JY0  16
    2 Z=(0,33333333*X)**2                                                      JY0  17
      J0=1,0-Z*(2,2499997-Z*(1,2656208-Z*(0,3163866-Z*(0,0444479-Z*(0,00JY0  18
     139444-0,00021*Z)))))                                                     JY0  19
      Y0=0,63661977*ALOG(0,5*X)*J0+0,36746691+Z*(0,60559366-Z*(0,7435038JY0  20
     14-Z*(0,25300117-Z*(0,04261214-Z*(0,00427916-0,00024846*Z)))))           JY0  21
      RETURN                                                                   JY0  22
    3 Z=3,0/X                                                                  JY0  23
      F=0,79788456-Z*(0,77E-6+Z*(0,0055274+Z*(0,00009512-Z*(0,00137237-ZJY0  24
     1*(0,00072805-Z*0,00014476*Z)))))                                         JY0  25
      P=0,78539816+Z*(0,04166397+Z*(0,00003954-Z*(0,00262573-Z*(0,000541JY0  26
     125+Z*(0,00029333-0,00013558*Z)))))                                       JY0  27
      Q=SQRT(1,0/X)                                                            JY0  28
      J0=Q*F*COS(X-P)                                                          JY0  29
      Y0=Q*F*SIN(X-P)                                                          JY0  30
    4 RETURN                                                                   JY0  31
      END                                                                      JY0  32-
      SUBROUTINE JY1(X,J1,Y1)                                                  JY1   1
C*************************************************************************** JY1   2
C                                                                              JY1   3
C     PURPOSE                                                                  JY1   4
C        COMPUTES BESSEL FUNCTIONS OF THE FIRST AND SECOND KIND,               JY1   5
C        FIRST ORDER, FOR POSITIVE ARGUMENTS,                                  JY1   6
C        SEE NBS AMS 55, P. 370.                                               JY1   7
C     DESCRIPTION OF PARAMETERS - ALL REAL                                     JY1   8
C        X- ARGUMENT, MUST BE >0                                               JY1   9
C        J1 - RETURNED FUNCTION VALUE, J1(X)                                   JY1  10
C        Y1 - RETURNED FUNCTION VALUE, Y1(X)                                   JY1  11
C                                                                              JY1  12
C*************************************************************************** JY1  13
      REAL J1                                                                  JY1  14
      IF (X-3,0) 1,2,3                                                         JY1  15
    1 IF (X) 4,4,2                                                             JY1  16
    2 Z=(0,33333333*X)**2                                                      JY1  17
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*—Continued

```
      J1=X*(0,5-Z*(0,56249985-Z*(0,21093573-Z*(0,03954289-Z*(0,00443319-JY1  18
     1Z*(0,00031761-0,00001109*Z))))))                                    JY1  19
      Y1=0,63661977*ALOG(0,5*X)*J1+(-0,6366198+Z*(0,2212091+Z*(2,1682709JY1  20
     1-Z*(1,3164827-Z*(0,3123951-Z*(0,0400976-0,0027873*Z))))))/X         JY1  21
      RETURN                                                              JY1  22
    3 Z=3,0/X                                                             JY1  23
      F=0,79788456+Z*(0,156E-5+Z*(0,01659667+Z*(0,00017105-Z*(0,00249511JY1  24
     1-Z*(0,00113653-0,00020033*Z)))))                                    JY1  25
      P=0,78539816-Z*(0,12499612+Z*(0,0000565-Z*(0,00637879-Z*(0,0007434JY1  26
     18+Z*(0,00079824-0,00029166*Z)))))                                   JY1  27
      Q=SQRT(1,0/X)                                                       JY1  28
      J1=Q*F*SIN(X-P)                                                     JY1  29
      Y1=-Q*F*COS(X-P)                                                    JY1  30
    4 RETURN                                                              JY1  31-
      END                                                                 JY1  32-
C*****************************************************************************FUA   1
C                                                                         FUA   2
C                                                                         FUA   3
C     PURPOSE                                                             FUA   4
C        COMPUTES FUNCTION VALUES OF F(UW,ALPHA)  =                       FUA   5
C        PAPADOPULOS,I.S. AND COOPER,H.H.,JR., 1967, DRAWDOWN IN          FUA   6
C        A WELL OF LARGE DIAMETER: WATER RESOURCES RESEARCH, V. 3,        FUA   7
C        NO. 1, P. 241-244.                                               FUA   8
C        PROGRAM BY S.S.PAPADOPULOS.                                      FUA   9
C     INPUT DATA = ONE OR MORE GROUPS, EACH GROUP CODED AS FOLLOWS        FUA  10
C        1 CARD = FORMAT (E10.5)                                          FUA  11
C             S = (ALPHA) = RW**2*S/RC**2 = RADIUS OF WELL (SCREEN        FUA  12
C                 OR OPEN BORE IN AQUIFER) SQUARED * STORAGE              FUA  13
C                 COEFFICIENT / RADIUS OF CASING (OVER INTERVAL OF        FUA  14
C                 WATER LEVEL CHANGE) SQUARED.                            FUA  15
C        1 CARD = FORMAT(16E5.0)                                          FUA  16
C             U= 16 VALUES OF UW = RW**2*S/(4*T*TIME) = RADIUS OF         FUA  17
C                 PUMPED WELL SQUARED * STORAGE COEFFICIENT /             FUA  18
C                 4 * TRANSMISSIVITY * TIME. IF LESS THAN 16 DESIRED,     FUA  19
C                 BLANK OR ZERO VALUES MAY BE CODED FOR THE REST.         FUA  20
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                       FUA  21
C        PEAK,SIMP,APEKE,EXBSL2,JY0,JY1 = MUST BE INCLUDED IN DECK.       FUA  22
C*****************************************************************************FUA  23
C                                                                         FUA  24
      COMMON XPK,YPK                                                      FUA  25
      COMMON/FBLK/A,B                                                     FUA  26
      EXTERNAL EXBSL2                                                     FUA  27
      DIMENSION U(16)                                                     FUA  28
      EPS=0,0001                                                          FUA  29
    1 READ (5,13,END=12) S                                                FUA  30
      IF (S) 1,1,2                                                        FUA  31
    2 READ (5,14) U                                                       FUA  32
      WRITE (6,15) S                                                      FUA  33
      DO 11 I=1,16                                                        FUA  34
      UW=U(I)                                                             FUA  35
      IF (UW) 1,1,3                                                       FUA  36
    3 B=0,25/UW                                                           FUA  37
      A=S+B                                                               FUA  38
      CALL APEKE(EXBSL2)                                                  FUA  39
      CALL PEAK(EXBSL2)                                                   FUA  40
      IF (XPK-1,0E-8) 4,5,5                                               FUA  41
    4 WRITE (6,16) UW,S,XPK,YPK                                           FUA  42
      GO TO 11                                                            FUA  43
    5 IF (XPK-1,0E8) 7,7,6                                                FUA  44
    6 WRITE (6,17) UW,S,XPK,YPK                                           FUA  45
      GO TO 11                                                            FUA  46
    7 HBAR=0,007*XPK
```

TABLE 8.2.—*Listing of programs for constant discharge from a fully penetrating well of finite diameter*—Continued

```
      CALL SIMPS(0,0,XPK,EPS,HBAR,SUM,DEL,EXBSL2)                    FUA  47
      X2=XPK                                                         FUA  48
      DX=XPK                                                         FUA  49
    8 DX=10,0*DX                                                     FUA  50
      X1=X2                                                          FUA  51
      X2=X1+DX                                                       FUA  52
      Y=EXBSL2(X2)                                                   FUA  53
      HBAR=0,007*DX                                                  FUA  54
      CALL SIMPS(X1,X2,EPS,HBAR,TRM,ERR,EXBSL2)                      FUA  55
      SUM=SUM+TRM                                                    FUA  56
      DEL=DEL+ERR                                                    FUA  57
      IF (X2-1,0E9) 9,10,10                                          FUA  58
    9 YT=1,5707963/X2**4                                             FUA  59
      IF (ABS(Y-YT)/YT-0,5E-6) 10,8,8                                FUA  60
   10 EST=0,52359878/X2**3                                           FUA  61
      SUM=SUM+EST                                                    FUA  62
      FUNS=3,2422779*S*S*SUM                                         FUA  63
      WRITE (6,18) UW,SUM,DEL,FUNS,XPK,YPK                           FUA  64
   11 CONTINUE                                                       FUA  65
      GO TO 1                                                        FUA  66
   12 STOP                                                           FUA  67
C                                                                    FUA  68
   13 FORMAT (E10,5)                                                 FUA  69
   14 FORMAT (16E5,0)                                                FUA  70
   15 FORMAT ('1','F(UW,ALPHA) FOR ALPHA=',1PE14,5/'0',7X,'UW',12X,'INTEFUA  71
     1GRAL',5X,'INTEGRAL ERROR',5X,'F(UW,ALPHA)',8X,'X(PEAK)',10X,'Y(PEAFUA  72
     2K)'/' ')                                                       FUA  73
   16 FORMAT (1H ,1PE14,7,9X,34HVALUES OF DUMMY VARIABLE TOO SMALL,1PE25FUA  74
     1,7,1PE17,7)                                                    FUA  75
   17 FORMAT (1H ,1PE14,7,9X,34HVALUES OF DUMMY VARIABLE TOO LARGE,1PE25FUA  76
     1,7,1PE17,7)                                                    FUA  77
   18 FORMAT (1H ,1PE14,5,1P5E17,5)                                  FUA  78
      END                                                            FUA  79-

      FUNCTION EXBSL2(X)                                             EB2   1
C*************************************************************************EB2   2
C                                                                    EB2   3
C     PURPOSE                                                        EB2   4
C        COMPUTES VALUES OF THE INTEGRAND FOR F(UW,ALPHA)            EB2   5
C     DESCRIPTION OF PARAMETER                                       EB2   6
C        X= REAL = ARGUMENT OF INTEGRAND                             EB2   7
C                                                                    EB2   8
C*************************************************************************EB2   9
      COMMON/PBLK/A,B                                                EB2  10
      IF (X) 1,1,2                                                   EB2  11
    1 EXBSL2=0,                                                      EB2  12
      GO TO 8                                                        EB2  13
    2 IF (X-1,E+7) 4,4,3                                             EB2  14
    3 EXBSL2=1,5707963/X**4                                          EB2  15
      GO TO 8                                                        EB2  16
    4 Y=B*X*X                                                        EB2  17
      IF (Y-,01) 5,5,6                                               EB2  18
    5 FNUM=Y*(1,-Y*(,5-Y*((1,/6,)-Y*(1,/24,))))                      EB2  19
      GO TO 7                                                        EB2  20
    6 FNUM=1,-EXP(-Y)                                                EB2  21
    7 CALL JY0(X,BJ0,BY0)                                            EB2  22
      CALL JY1(X,BJ1,BY1)                                            EB2  23
      DEN=((X*BJ0-A*BJ1)**2+(X*BY0-A*BY1)**2)*X**3                   EB2  24
      EXBSL2=FNUM/DEN                                                EB2  25
    8 RETURN                                                         EB2  26
      END                                                            EB2  27-
```

TABLE 9.2.—*Listing of program to compute change in water level due to sudden injection of a slug of water into a well*

```
C*************************************************************************** FBA   1
C                                                                             FBA   2
C     PURPOSE                                                                 FBA   3
C        COMPUTES FUNCTION VALUES OF F(BETA,ALPHA) = THE SLUG TEST            FBA   4
C        FUNCTION = COOPER,H,H.,JR.,, BREDEHOEFT,J,D.,, AND PAPADOPULOS,      FBA   5
C        I,S.,, 1967, RESPONSE OF A FINITE-DIAMETER WELL TO AN                FBA   6
C        INSTANTANEOUS CHARGE OF WATER; WATER RESOURCES RESEARCH,             FBA   7
C        V, 3, NO. 1, P. 263-269.                                             FBA   8
C        PROGRAM BY S.S.PAPADOPULOS.                                          FBA   9
C     INPUT DATA                                                              FBA  10
C        1 OR MORE CARDS - FORMAT(F16.5)                                      FBA  11
C             A = (ALPHA) = RW**2*S/RC**2 = RADIUS OF WELL (SCREEN OR         FBA  12
C                 OPEN BORE IN AQUIFER) SQUARED * STORAGE COEFFICIENT         FBA  13
C                 / RADIUS OF CASING (OVER INTERVAL OF WATER LEVEL            FBA  14
C                 CHANGE) SQUARED.                                            FBA  15
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                           FBA  16
C        PRX,DJY0,DJY1,DSIMPS - MUST BE INCLUDED IN DECK                      FBA  17
C                                                                             FBA  18
C     METHOD                                                                  FBA  19
C        THIS PROGRAM CALCULATES THE SLUG TEST FUNCTION, F(BETA,ALPHA),       FBA  20
C        FOR VALUES OF BETA RANGING FROM 0.001 TO 1000.0 BY INCREMENTING      FBA  21
C        BETA ACCORDING TO DATA ARRAY BB(I), AVERAGE COMPUTATION TIME         FBA  22
C        IS ABOUT 30 SECONDS PER VALUE OF ALPHA ON IBM 360/155.               FBA  23
C                                                                             FBA  24
C*************************************************************************** FBA  25
      DOUBLE PRECISION A,B,PI,ZZ,EPS,Y,X1,X2,TERM,FAB,DATAN,DEL,HBAR          FBA  26
      DIMENSION ZZ(40), BB(39)                                                FBA  27
      COMMON A,B,PI                                                           FBA  28
      EXTERNAL PRX                                                            FBA  29
      DATA ZZ/0.D+0,1.D-10,1.D-9,1.D-8,1.D-7,1.D-6,1.D-5,1.D-4,               FBA  30
     1   1.D-3,1.D-2,1.D-1,1.D+0,2.D+0,3.D+0,4.D+0,5.D+0,6.D+0,7.D+0,8.D+0,   FBA  31
     2   9.D+0,1.D+1,2.D+1,3.D+1,4.D+1,5.D+1,6.D+1,7.D+1,8.D+1,               FBA  32
     3   9.D+1,1.D+2,1.25D+2,1.5D+2/                                         FBA  33
      DATA BB/.001,.002,.004,.006,.008,.01,.02,.04,.06,.08,.1,.2,.4,.6,.8,1.,FBA  34
     1.,2.,3.,4.,5.,6.,7.,8.,9.,10.,20.,30.,40.,50.,60.,70.,80.,90.,100.,   FBA  35
     2200.,200.,400.,600.,800.,1000./                                        FBA  36
      PI=4.*DATAN(1.0D+00)                                                    FBA  37
      EPS=0.00001                                                             FBA  38
    1 READ (5,6) A                                                            FBA  39
      IF (A.LE.0.0) GO TO 5                                                   FBA  40
      WRITE (6,7) A                                                           FBA  41
      WRITE (6,8)                                                             FBA  42
      DO 4 I=1,39                                                             FBA  43
      B=BB(I)                                                                 FBA  44
      Y=0.0                                                                   FBA  45
      DO 2 L=1,39                                                             FBA  46
      X1=ZZ(L)                                                                FBA  47
      X2=ZZ(L+1)                                                              FBA  48
      HBAR=0.                                                                 FBA  49
      CALL DSIMPS(X1,X2,EPS,HBAR,TERM,DEL,PRX)                                FBA  50
      Y=Y+TERM                                                                FBA  51
      IF (L.GT.20.AND.TERM.LT.EPS) GO TO 3                                    FBA  52
    2 CONTINUE                                                                FBA  53
    3 FAB=4.*A*Y/(PI*PI)                                                      FBA  54
    4 WRITE (6,9) B,FAB                                                       FBA  55
      GO TO 1                                                                 FBA  56
    5 STOP                                                                    FBA  57
C                                                                             FBA  58
C                                                                             FBA  59
    6 FORMAT (F16.5)                                                          FBA  60
    7 FORMAT ('1',41X,'F(BETA,ALPHA) FOR ALPHA=',1PD9.2/)                     FBA  61
    8 FORMAT ('0',53X,'BETA',13X,'H/H0'/)                                     FBA  62
    9 FORMAT (' ',51X,1PD8.2,10X,0PF6.4)                                      FBA  63
      END                                                                     FBA  64
```

TABLE 9.2.—*Listing of program to compute change in water level due to sudden injection of a slug of water into a well*—Continued

```
      DOUBLE PRECISION FUNCTION PRX(X)                                          PRX    1
C*************************************************************************** PRX    2
C                                                                             PRX    3
C     PURPOSE                                                                 PRX    4
C        COMPUTE VALUES OF THE INTEGRAND FOR F(BETA,ALPHA)                    PRX    5
C     DESCRIPTION OF PARAMETER                                                PRX    6
C        X = DOUBLE PRECISION = ARGUMENT OF INTEGRAND                         PRX    7
C                                                                             PRX    8
C*************************************************************************** PRX    9
      DOUBLE PRECISION A,B,PI,XX,X,C,F1,F2,J0,Y0,J1,Y1                        PRX   10
      DOUBLE PRECISION DLOG,DSQRT,DEXP                                        PRX   11
      COMMON A,B,PI                                                           PRX   12
      XX=DSQRT(A*X/B)                                                         PRX   13
      IF (X) 6,1,2                                                            PRX   14
    1 PRX=(PI*PI)/(16.*A*B)                                                   PRX   15
      GO TO 6                                                                 PRX   16
    2 IF (X.LT.150.) GO TO 3                                                  PRX   17
      PRX=0.0                                                                 PRX   18
      GO TO 6                                                                 PRX   19
    3 IF (XX.GT.0.0001) GO TO 4                                               PRX   20
      C=DEXP(5.772156649D-01)/2.                                              PRX   21
      F1=PI*X*(1.-A)                                                          PRX   22
      F2=X*DLOG(C*C*A*X/8)+4.*B                                               PRX   23
      PRX=(B*PI*PI*DEXP(-X))/(A*(F1*F1+F2*F2))                                PRX   24
      GO TO 6                                                                 PRX   25
    4 IF (XX.LT.50.) GO TO 5                                                  PRX   26
      PRX=(PI*DEXP(-X))/(2.*XX*(X+4.*A*B))                                    PRX   27
      GO TO 6                                                                 PRX   28
    5 CALL DJY0(XX,J0,Y0)                                                     PRX   29
      CALL DJY1(XX,J1,Y1)                                                     PRX   30
      F1=(XX*J0-2.*A*J1)                                                      PRX   31
      F2=(XX*Y0-2.*A*Y1)                                                      PRX   32
      PRX=DEXP(-X)/(X*(F1*F1+F2*F2))                                          PRX   33
    6 RETURN                                                                  PRX   34
      END                                                                     PRX   35

      SUBROUTINE DJY0(X,J0,Y0)                                                DJ0    1
C*************************************************************************** DJ0    2
C                                                                             DJ0    3
C     PURPOSE                                                                 DJ0    4
C        COMPUTES BESSEL FUNCTIONS OF THE FIRST AND SECOND KIND,              DJ0    5
C        ZERO ORDER, FOR POSITIVE ARGUMENTS,                                  DJ0    6
C     DESCRIPTION OF PARAMETERS - ALL DOUBLE PRECISION                        DJ0    7
C        X- ARGUMENT, MUST BE >0                                              DJ0    8
C        J0 - RETURNED FUNCTION VALUE, J0(X)                                  DJ0    9
C        Y0 - RETURNED FUNCTION VALUE, Y0(X)                                  DJ0   10
C                                                                             DJ0   11
C*************************************************************************** DJ0   12
      DOUBLE PRECISION Z,J0,Y0,F,P,Q,U,W,X,DLOG,DCOS,DSIN,DSQRT               DJ0   13
      IF (X-3.0) 1,2,3                                                        DJ0   14
    1 IF (X) 4,4,2                                                            DJ0   15
    2 Z=(X/3.0)**2                                                            DJ0   16
      J0=1.0-Z*(2.2499997-Z*(1.2656208-Z*(0.3163866-Z*(0.0444479-Z*(0.00    DJ0   17
     139444-0.00021*Z)))))                                                    DJ0   18
      W=(0.5D0)*X                                                             DJ0   19
      Y0=0.63661977*DLOG(W)*J0+0.36746691+Z*(0.60559366-Z*(0.74350384-Z*Z    DJ0   20
     1(0.25300117-Z*(0.04261214-Z*(0.00427916-0.00024846*Z)))))               DJ0   21
      RETURN                                                                  DJ0   22
    3 Z=3.0/X                                                                 DJ0   23
      F=0.79788456-Z*(0.77D-6+Z*(0.0055274+Z*(0.0009512-Z*(0.00137237-ZDJ0   24
     1*(0.00072805-0.00014476*Z)))))                                          DJ0   25
      P=0.78539816+Z*(0.04166397+Z*(0.00003954-Z*(0.00262573-Z*(0.000541DJ0   26
     125+Z*(0.00029333-0.00013558*Z)))))                                      DJ0   27
```

TABLE 9.2.—*Listing of program to compute change in water level due to sudden injection of a slug of water into a well*—Continued

```
      U=(1.0D0)/X                                                        DJ0  28
      Q=DSQRT(U)                                                         DJ0  29
      J0=Q*F*DCOS(X-P)                                                   DJ0  30
      Y0=Q*F*DSIN(X-P)                                                   DJ0  31
    4 RETURN                                                             DJ0  32
      END                                                                DJ0  33
      SUBROUTINE DJY1(X,J1,Y1)                                           DJ1   1
C*******************************************************************************DJ1   2
C                                                                        DJ1   3
C     PURPOSE                                                            DJ1   4
C        COMPUTES BESSEL FUNCTIONS OF THE FIRST AND SECOND KIND,         DJ1   5
C        FIRST ORDER, FOR POSITIVE ARGUMENTS.                            DJ1   6
C     DESCRIPTION OF PARAMETERS - ALL DOUBLE PRECISION                   DJ1   7
C        X- ARGUMENT, MUST BE >0                                         DJ1   8
C        J1 - RETURNED FUNCTION VALUE, J1(X)                             DJ1   9
C        Y1 - RETURNED FUNCTION VALUE, Y1(X)                             DJ1  10
C                                                                        DJ1  11
C*******************************************************************************DJ1  12
      DOUBLE PRECISION X,J1,Y1,Z,W,DLOG,F,P,U,Q,DSQRT,DSIN,DCOS          DJ1  13
      IF (X-3.0) 1,2,3                                                   DJ1  14
    1 IF (X) 4,4,2                                                       DJ1  15
    2 Z=(X/3.0)**2                                                       DJ1  16
      J1=X*(0.5-Z*(0.56249985-Z*(0.21093573-Z*(0.03954289-Z*(0.00443319-DJ1  17
     1Z*(0.00031761-0.00001109*Z))))))                                   DJ1  18
      W=(0.5D0)*X                                                        DJ1  19
      Y1=0.63661977*DLOG(W)*J1+(-0.6366198+Z*(0.2212091+Z*(2.1682709-Z*(DJ1  20
     11.3164827-Z*(0.3123951-Z*(0.0400976-0.0027873*Z))))))/X            DJ1  21
      RETURN                                                             DJ1  22
    3 Z=3.0/X                                                            DJ1  23
      F=0.79788456+Z*(0.156D-5+Z*(0.01659667+Z*(0.00017105-Z*(0.00249511DJ1  24
     1-Z*(0.00113653-0.00020033*Z)))))                                   DJ1  25
      P=0.78539816-Z*(0.12499612+Z*(0.0000565-Z*(0.00637879-Z*(0.0007434DJ1  26
     18+Z*(0.00079824-0.00029166*Z)))))                                  DJ1  27
      U=(1.0D0)/X                                                        DJ1  28
      Q=DSQRT(U)                                                         DJ1  29
      J1=Q*F*DSIN(X-P)                                                   DJ1  30
      Y1=-Q*F*DCOS(X-P)                                                  DJ1  31
    4 RETURN                                                             DJ1  32
      END                                                                DJ1  33
      SUBROUTINE DSIMPS(A,B,EPS,HBAR,AREA,DEL,F)                         DSI   1
C*******************************************************************************DSI   2
C                                                                        DSI   3
C     PURPOSE                                                            DSI   4
C        TO DETERMINE THE INTEGRAL OF A FUNCTION, F, FROM A TO B,        DSI   5
C        USING SIMPSON'S RULE.                                           DSI   6
C     DESCRIPTION OF PARAMETERS                                          DSI   7
C        ALL DOUBLE PRECISION                                            DSI   8
C        A - LOWER LIMIT OF INTEGRAL                                     DSI   9
C        B - UPPER LIMIT OF INTEGRAL                                     DSI  10
C        EPS - DESIRED ACCURACY                                          DSI  11
C        HBAR - MINIMUM DIVISION OF THE INTERVAL                         DSI  12
C        AREA - COMPUTED VALUE OF INTEGRAL BETWEEN Q AND R               DSI  13
C        DEL - COMPUTED ESTIMATE OF ERROR                                DSI  14
C        F- THE INTEGRAND (FUNCTION REFERENCE)                           DSI  15
C     METHOD                                                             DSI  16
C        USES SIMPSON'S RULE TO COMPUTE A SUM APPROXIMATING THE INTEGRAL DSI  17
C        USES INITIAL H=(B-A)/2, COMPUTES A SEQUENCE OF SUMS BY HALVING  DSI  18
C        H EACH TIME, COMPUTES ESTIMATE OF ERROR (DEL) AS (PREVIOUS      DSI  19
C        SUM - CURRENT SUM)/15, COMPUTATION STOPS WHEN 1) H<HBAR,        DSI  20
C        2) ABS(DEL)<ABS(EPS*CURRENT SUM), IF HBAR IS LE 0,              DSI  21
C        THEN HBAR=.007*(B-A),                                           DSI  22
C                                                                        DSI  23
```

TABLE 9.2.—*Listing of program to compute change in water level due to sudden injection of a slug of water into a well*—Continued

```
C*****************************************************************************DSI   24
      DOUBLE PRECISION H,HBAR,AREA,DEL,S1,S2,S3,SC,SP,X,A,B,EPS,F,DABS          DSI   25
C  AREA OF F FROM A TO B,EPS IS DESIRED ACCURACY, HBAR THE MINIMUM               DSI   26
C  ALLOWABLE INTERVAL, DEL THE ESTIMATE OF THE ERROR                            DSI   27
      H=B-A                                                                     DSI   28
      IF (H) 1,1,2                                                              DSI   29
    1 AREA=0.0                                                                  DSI   30
      DEL=0.0                                                                   DSI   31
      GO TO 10                                                                  DSI   32
    2 SP=1.0D35                                                                 DSI   33
      S3=0.0                                                                    DSI   34
      S1=F(A)+F(B)                                                              DSI   35
      IF (HBAR) 3,3,4                                                           DSI   36
    3 HBAR=0.007*H                                                              DSI   37
    4 S2=0.0                                                                    DSI   38
      X=A+0.5*H                                                                 DSI   39
    5 S2=S2+4.0*F(X)                                                            DSI   40
      X=X+H                                                                     DSI   41
      IF (X-B) 5,5,6                                                            DSI   42
    6 SC=(S1+S2+S3)*H*0.16666666667                                             DSI   43
      DEL=0.06666666667*(SP-SC)                                                 DSI   44
      IF (DABS(DEL)-DABS(EPS*SC)) 7,8,8                                         DSI   45
    7 AREA=SC-DEL                                                               DSI   46
      GO TO 10                                                                  DSI   47
    8 S3=S3+0.5*S2                                                              DSI   48
      H=0.5*H                                                                   DSI   49
      IF (H-HBAR) 7,9,9                                                         DSI   50
    9 SP=SC                                                                     DSI   51
      GO TO 4                                                                   DSI   52
   10 RETURN                                                                    DSI   53
      END                                                                       DSI   54
```

TABLE 11.1.—*Listing of program to compute the convolution integral for a leaky aquifer*

```
C*****************************************************************************HRT    1
C                                                                               HRT    2
C     PURPOSE                                                                   HRT    3
C        COMPUTES CHANGES IN WATER LEVEL, H(R,T), IN RESPONSE TO                HRT    4
C        VARYING DISCHARGE USING THE CONVOLUTION INTEGRAL FOR                   HRT    5
C        LEAKY AQUIFERS - EQ. 3 OF HOENCH,ALLEN,1971, GROUND-WATER              HRT    6
C        FLUCTUATIONS IN RESPONSE TO ARBITRARY PUMPAGE: GROUND WATER,           HRT    7
C        V.9, NO.2,P.4-8.                                                       HRT    8
C     INPUT DATA - ONE OR MORE GROUPS, EACH GROUP CODED AS FOLLOWS              HRT    9
C        1 CARD - FORMAT(2E10.5,4X,I1,5X,E10.5)                                 HRT   10
C             TBEGIN - SMALLEST VALUE OF TIME FOR OUTPUT.                       HRT   11
C             TEND - LARGEST VALUE OF TIME FOR OUTPUT.                          HRT   12
C             IQ - INDICATES FORM OF DISCHARGE FUNCTION, Q(T).                  HRT   13
C                  IQ=1,2,3 REFER TO DISCHARGE FUNCTIONS IN                     HRT   14
C                  HANTUSH,M.S., 1964, HYDRAULICS OF WELLS IN CHOW,             HRT   15
C                  VEN TE, ED., ADVANCES IN HYDROSCIENCE, VOL. 1:               HRT   16
C                  ACADEMIC PRESS INC., NEW YORK, P. 281-442.                   HRT   17
C                  IQ=1, Q(T) IS AN EXPONENTIAL FUNCTION, CASE A,               HRT   18
C                       P. 343 OF HANTUSH.                                      HRT   19
C                  IQ=2, Q(T) IS A HYPERBOLIC FUNCTION, CASE B,                 HRT   20
C                       P. 344 OF HANTUSH.                                      HRT   21
C                  IQ=3, Q(T) IS AN INVERSE SQUARE ROOT FUNCTION,               HRT   22
C                       CASE C, P. 344 OF HANTUSH.                              HRT   23
```

TABLE 11.1.—*Listing of program to compute the convolution integral for a leaky aquifer*—Continued

```
C                                                                       HRT  24
C           IQ=4, Q(T) IS A FIFTH-DEGREE POLYNOMIAL,                    HRT  25
C           IQ=5, Q(T) IS A PIECEWISE LINEAR FUNCTION OF                HRT  26
C                 TIME (EIGHT SEGMENTS),                                HRT  27
C       QR = REFERENCE DISCHARGE, ZERO OR BLANK FOR PROJECTION,         HRT  28
C   1 OR 4 CARDS, DEPENDING ON IQ,                                      HRT  29
C       IF IQ=1,2,3 - 1 CARD - FORMAT(3E10.3)                           HRT  30
C           QST = EVENTUAL CONSTANT DISCHARGE,                          HRT  31
C           DELTA = RATE PARAMETER,                                     HRT  32
C           TSTAR = TIME PARAMETER,                                     HRT  33
C       IF IQ=4 - 1 CARD - FORMAT(6E10.3)                               HRT  34
C           AQ(6) = 6 VALUES = THE POLYNOMIAL COEFFICIENTS              HRT  35
C                 WITH A0 FIRST AND A5 LAST,                            HRT  36
C       IF IQ=5 - 4 CARDS - FORMAT(6E10.3)                              HRT  37
C           TI(I),AI(I),BI(I),TI(I+1),AI(I+1),BI(I+1),I=1,3,5,7         HRT  38
C           PARAMETERS OF THE PIECEWISE LINEAR FUNCTION                 HRT  39
C           (8 SEGMENTS), CODED 2 SEGMENTS PER CARD, FIRST              HRT  40
C           AND SECOND SEGMENTS ON FIRST CARD, THEN SEQUENTIALLY        HRT  41
C           ON SUCCEEDING CARDS, EACH SEGMENT HAS THREE                 HRT  42
C           PARAMETERS WHICH ARE IN CODING ORDER                        HRT  43
C                TI = ENDING TIME OF THE SEGMENT,                       HRT  44
C                AI = DISCHARGE AT BEGINNING OF SEGMENT,                HRT  45
C                BI = RATE OF CHANGE IN DISCHARGE DURING SEG,           HRT  46
C           THE DISCHARGE FUNCTION IN EACH SEGMENT HAS THE              HRT  47
C           FORM Q(T) = AI(I)+BI(I)*(T-TI(I-1)), IF LESS THAN 8         HRT  48
C           SEGMENTS ARE NEEDED, BLANKS CAN BE CODED FOR                HRT  49
C           SUCCEEDING SEGMENTS,                                        HRT  50
C     2 OR MORE CARDS - FORMAT(4E10.3)                                  HRT  51
C       R = RADIAL DISTANCE FROM PUMPED WELL, BLANK OR ZERO             HRT  52
C           SIGNALS PROGRAM AS END TO GROUP OF DATA,                    HRT  53
C       S = STORAGE COEFFICIENT                                         HRT  54
C       T = TRANSMISSIVITY                                              HRT  55
C       PM = (P'/M') = HYD. COND. OF CONFINING BED DIVIDED              HRT  56
C           BY THICKNESS OF CONFINING BED,                              HRT  57
C   SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                       HRT  58
C     CONVOL,Q - MUST BE INCLUDED IN DECK,                              HRT  59
C***********************************************************************HRT  60
      DIMENSION       D(12),IEX(12),X(6),H(12,6),QS(12,6),CP(12),CT(12) HRT  61
      DIMENSION H1(12),H2(12),Q1(12),Q2(12)                             HRT  62
      DIMENSION H3(12),H4(12),Q3(12),Q4(12)                             HRT  63
      COMMON AQ(6),TI(9),AI(9),BI(9),QST,DELTA,TSTAR                    HRT  64
      DATA CP/12*'  T*'/,CT/12*'1/U*'/,D/12*'10**'/                     HRT  65
      DATA H1/12*'  S('/,H2/12*'R,T)'/,Q1/12*'     '/,Q2/12*'Q(T)'/     HRT  66
      DATA H3/12*'   S'/,H4/12*'D(T)'/,Q3/12*' Q(T'/,Q4/12*')'/QR'/     HRT  67
      DATA X/1.,.1,.5,.2,.3,.5,.7,/                                     HRT  68
      TI(1)=0.                                                          HRT  69
      N=500                                                             HRT  70
    1 READ (5,18,END=17) TBEGIN,TEND,IQ,QR                              HRT  71
      IF (IQ.LT.4) READ (5,19) QST,DELTA,TSTAR                          HRT  72
      IF (IQ.EQ.4) READ (5,19) AQ                                       HRT  73
      IF (IQ.EQ.5) READ (5,19) (TI(I),AI(I),BI(I),I=2,9)                HRT  74
      WRITE (6,24)                                                      HRT  75
    2 READ (5,19) R,S,T,PM                                              HRT  76
      IF (R.EQ.0.) GO TO 1                                              HRT  77
      A=R*R*S/(4.*T)                                                    HRT  78
      B=PM/S                                                            HRT  79
      Y=ALOG10(TBEGIN)                                                  HRT  80
```

TABLE 11.1.—*Listing of program to compute the convolution integral for a leaky aquifer*—Continued

```
      IF (Y) 3,5,4                                                    HRT  81
    3 Y=Y-.001                                                        HRT  82
      GO TO 5                                                         HRT  83
    4 Y=Y+.001                                                        HRT  84
    5 IBEGIN=Y                                                        HRT  85
      Y=ALOG10(TEND)                                                  HRT  86
      IF (Y) 6,8,7                                                    HRT  87
    6 Y=Y-.001                                                        HRT  88
      GO TO 8                                                         HRT  89
    7 Y=Y+.001                                                        HRT  90
    8 IEND=Y                                                          HRT  91
      M=IEND-IBEGIN+1                                                 HRT  92
      IF (M.GT.12) M=12                                               HRT  93
      DO 10 I=1,M                                                     HRT  94
      IEX(I)=IBEGIN+I-1                                               HRT  95
      Y=10.**(IBEGIN+I-1)                                             HRT  96
      DO 10 J=1,6                                                     HRT  97
      TIME=X(J)*Y                                                     HRT  98
      IF (QR.GT.0.) TIME=A*TIME                                       HRT  99
      CALL CONVOL(TIME,A,B,N,IQ,SUM)                                  HRT 100
      IF (QR.GT.0.) GO TO 9                                           HRT 101
      H(I,J)=SUM/(12.5664*T)                                          HRT 102
      QS(I,J)=Q(TIME,IQ)                                              HRT 103
      GO TO 10                                                        HRT 104
    9 H(I,J)=SUM/QR                                                   HRT 105
      QS(I,J)=Q(TIME,IQ)/QR                                           HRT 106
   10 CONTINUE                                                        HRT 107
      K=M                                                             HRT 108
      IF (M.GT.6) K=6                                                 HRT 109
      IF (QR.GT.0.) GO TO 11                                          HRT 110
      WRITE (6,20) A,B,(CP(I),D(I),IEX(I),I=1,K)                      HRT 111
      WRITE (6,21) (H1(I),H2(I),Q1(I),Q2(I),I=1,K)                    HRT 112
      GO TO 12                                                        HRT 113
   11 WRITE (6,25) A,B,QR,(CT(I),D(I),IEX(I),I=1,K)                   HRT 114
      WRITE (6,21) (H3(I),H4(I),Q3(I),Q4(I),I=1,K)                    HRT 115
   12 DO 13 J=1,6                                                     HRT 116
      WRITE (6,22) X(J),(H(I,J),QS(I,J),I=1,K)                        HRT 117
   13 CONTINUE                                                        HRT 118
      IF (M.LE.6) GO TO 2                                             HRT 119
      K1=K+1                                                          HRT 120
      IF (QR.GT.0.) GO TO 14                                          HRT 121
      WRITE (6,23) (CP(I),D(I),IEX(I),I=K1,M)                         HRT 122
      WRITE (6,21) (H1(I),H2(I),Q1(I),Q2(I),I=K1,M)                   HRT 123
      GO TO 15                                                        HRT 124
   14 WRITE (6,26) (CT(I),D(I),IEX(I),I=K1,M)                         HRT 125
      WRITE (6,21) (H3(I),H4(I),Q3(I),Q4(I),I=K1,M)                   HRT 126
   15 DO 16 J=1,6                                                     HRT 127
      WRITE (6,22) X(J),(H(I,J),QS(I,J),I=K1,M)                       HRT 128
   16 CONTINUE                                                        HRT 129
      GO TO 2                                                         HRT 130
   17 STOP                                                            HRT 131
C                                                                     HRT 132
   18 FORMAT (2E10.5,4X,I1,5X,E10.5)                                  HRT 133
   19 FORMAT (6E10.3)                                                 HRT 134
   20 FORMAT ('0','R**2*S/(4*TRANS)=',1PE10.3,', K'''/(S*B'')=',E10.3,/'0'HRT 135
     1,2X,'T',5X,6(2A4,I2,9X))                                        HRT 136
   21 FORMAT (' ',4X,6(2A4,2X,2A4,1X))                                HRT 137
```

TABLE 11.1.—*Listing of program to compute the convolution integral for a leaky aquifer*—Continued

```
   22 FORMAT (' ',F4.1,6(0PF8.3,1PE11.3))                                  HRT 138
   23 FORMAT ('0',2X,'T',5X,6(2A4,I2,9X))                                  HRT 139
   24 FORMAT (1H1)                                                         HRT 140
   25 FORMAT ('0','R**2*S/(4*TRANS)=',1PE10.3,', K'/(S*B')=',E10.3,',      HRT 141
     1QR=',E10.3/'0',1X,'1/U',4X,6(2A4,I2,9X))                             HRT 142
   26 FORMAT ('0',1X,'1/U',4X,6(2A4,I2,9X))                                HRT 143
      END                                                                  HRT 144*
      SUBROUTINE CONVOL(TIME,A,B,N,IQ,SUM)                                 CON   1
C*************************************************************************** CON   2
C                                                                          CON   3
C     PURPOSE                                                              CON   4
C        COMPUTES VALUES OF THE CONVOLUTION INTEGRAL FOR LEAKY             CON   5
C        AQUIFERS, THE INTEGRAL IS, FROM 0 TO T, OF                        CON   6
C        Q(T-T')/T'*EXP(-A/T'-B*T')*DT',                                   CON   7
C     DESCRIPTION OF PARAMETERS                                            CON   8
C        A,B,SUM ARE REAL; N,IQ ARE INTEGER,                               CON   9
C        A = R**2*S/(4*T) = RADIAL DISTANCE SQUARED * STORAGE              CON  10
C             COEFFICIENT / 4 * TRANSMISSIVITY,                            CON  11
C        B = P'/(S*M') = HYD. COND. OF CONFINING BED DIVIDED BY            CON  12
C             AQUIFER STORAGE COEFFICIENT * THICKNESS OF CONF. BED,        CON  13
C        N = NUMBER OF INCREMENTS FOR EACH INTERVAL OF THE SUM,            CON  14
C        IQ = INDICATES FORM OF DISCHARGE FUNCTION,                        CON  15
C     SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED                        CON  16
C        Q                                                                 CON  17
C     METHOD                                                               CON  18
C        APPROXIMATES INTEGRAL BY SUMMING THE TRAPEZOIDAL RULE APPLIED     CON  19
C        TO A SEQUENCE OF SEGMENTS, LOWER LIMIT OF FIRST SEGMENT IS        CON  20
C        PICKED AT POINT WHERE EXPONENT > -100 ,                           CON  21
C        IF SUCH A POINT DOES NOT EXIST (A*B > 2500) A FUNCTION VALUE      CON  22
C        OF 0 IS RETURNED, UPPER LIMIT = 10 * LOWER LIMIT FOR EACH         CON  23
C        SEGMENT, USES INCREMENT OF DELTA T' = (U-L)/N WHERE N IS THE      CON  24
C        NUMBER OF INCREMENTS IN THE CALL, CEASES SUMMATION WHEN           CON  25
C        EXPONENT < -101 .                                                 CON  26
C                                                                          CON  27
C*************************************************************************** CON  28
      REAL*8 DSUM                                                          CON  29
      REAL*4 NEWT,NEWTP,NEWX,NEWF                                          CON  30
      DSUM=0.D+0                                                           CON  31
      IS=0                                                                 CON  32
C     INITIAL T' COMPUTED FROM A,B                                         CON  33
      AB=A*B                                                               CON  34
      IF (AB.GE.2500.) GO TO 7                                             CON  35
      IF (B.GT.0.) GO TO 2                                                 CON  36
    1 OLDT=.01*A                                                           CON  37
      GO TO 3                                                              CON  38
    2 OLDT=(1.-SQRT(1.-AB/2500.))*50./B                                    CON  39
      IF (OLDT.EQ.0.) GO TO 1                                              CON  40
C     INITIAL T-T'                                                         CON  41
    3 OLDTP=TIME-OLDT                                                      CON  42
      OLDX=-A/OLDT-B*OLDT                                                  CON  43
      OLDF=Q(OLDTP,IQ)*EXP(OLDX)/OLDT                                      CON  44
C     END OF SUMMATION SEGMENT IS 10 TIMES THE BEGINNING                   CON  45
    4 ENDT=10.*OLDT                                                        CON  46
      IF (ENDT.LT.TIME) GO TO 5                                            CON  47
      IF (OLDT.GE.TIME) GO TO 7                                            CON  48
      IS=1                                                                 CON  49
      ENDT=TIME                                                            CON  50
```

TABLE 11.1.—*Listing of program to compute the convolution integral for a leaky aquifer*—Continued

```
C     DELTA T' IS COMPUTED FROM LENGTH AND NUMBER OF INCREMENTS                CON   51
    5 DELT=(ENDT-OLDT)/N                                                       CON   52
      DO 6 I=1,N                                                               CON   53
C     T' IS INCREMENTED BY DELTA T'                                            CON   54
      NEWT=OLDT+DELT                                                           CON   55
      NEWX=-A/NEWT-B*NEWT                                                      CON   56
C     TERMINATES SUMMATION WHEN EXP(-A/T'-B*T') < 1.37E-44                     CON   57
      IF (NEWX.LT.-101.) GO TO 7                                               CON   58
      NEWTP=TIME-NEWT                                                          CON   59
      NEWF=Q(NEWTP,IQ)*EXP(NEWX)/NEWT                                          CON   60
      DSUM=DSUM+(NEWF+OLDF)*DELT                                               CON   61
      OLDT=NEWT                                                                CON   62
      OLDF=NEWF                                                                CON   63
    6 CONTINUE                                                                 CON   64
      IF (IS.GT.0) GO TO 7                                                     CON   65
C     IF T' < T, BEGINS A NEW SEGMENT                                          CON   66
      GO TO 4                                                                  CON   67
    7 SUM=DSUM/2.0+0                                                           CON   68
      RETURN                                                                   CON   69
      END                                                                      CON   70

      FUNCTION Q(TIME,IQ)                                                      Q     1
C*************************************************************************   Q     2
C                                                                              Q     3
C     PURPOSE                                                                  Q     4
C        COMPUTES THE DISCHARGE FUNCTION, Q(T)                                 Q     5
C     DESCRIPTION OF PARAMETERS                                                Q     6
C        TIME = REAL = ELAPSED TIME SINCE BEGINNING OF DISCHARGE,              Q     7
C        IQ = INTEGER = INDICATES FORM OF DISCHARGE FUNCTION,                  Q     8
C             IQ=1,2,3, CASES A,B,C, RESPECTIVELY, OF HANTUSH,M.S.,            Q     9
C                  1964, HYDRAULICS OF WELLS IN CHOW, VEN TE, ED.,             Q    10
C                  ADVANCES IN HYDROSCIENCE, VOL. 1: ACADEMIC PRESS,           Q    11
C                  NEW YORK, P. 343,344,                                       Q    12
C             IQ=4, DISCHARGE IS A FIFTH DEGREE POLYNOMIAL OF TIME,            Q    13
C             IQ=5, DISCHARGE IS A PIECEWISE LINEAR FUNCTION OF UP TO          Q    14
C                  8 SEGMENTS,                                                 Q    15
C     METHOD                                                                   Q    16
C        FORTRAN EVALUATION OF FUNCTIONS,                                      Q    17
C                                                                              Q    18
C*************************************************************************   Q    19
      COMMON AQ(6),TI(9),AI(9),BI(9),QST,DELTA,TSTAR                           Q    20
      GO TO (1,2,3,4,5), IQ                                                    Q    21
    1 Q=QST*(1.+DELTA*EXP(-TIME/TSTAR))                                        Q    22
      RETURN                                                                   Q    23
    2 Q=QST*(1.+DELTA/(1.+TIME/TSTAR))                                         Q    24
      RETURN                                                                   Q    25
    3 Q=QST*(1.+DELTA/SQRT(1.+TIME/TSTAR))                                     Q    26
      RETURN                                                                   Q    27
    4 Q=AQ(1)+TIME*(AQ(2)+TIME*(AQ(3)+TIME*(AQ(4)+TIME*(AQ(5)+TIME*AQ(6)       Q    28
     1))))                                                                     Q    29
      RETURN                                                                   Q    30
    5 DO 6 I=2,9                                                               Q    31
      IF (TIME.LE.TI(I)) GO TO 7                                               Q    32
    6 CONTINUE                                                                 Q    33
      I=9                                                                      Q    34
    7 Q=AI(I)+BI(I)*(TIME-TI(I-1))                                             Q    35
      RETURN                                                                   Q    36
      END                                                                      Q    37
```

UNITED STATES DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

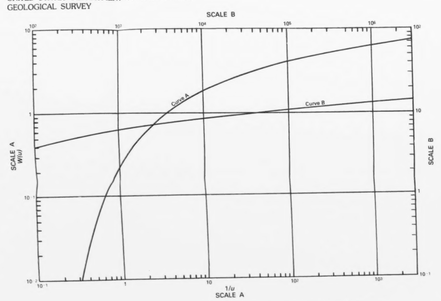

FIGURE 1.2.—Type curve of dimensionless drawdown ($W(u)$) versus dimensionless time ($1/u$) for constant discharge from an artesian well (Theis curve).

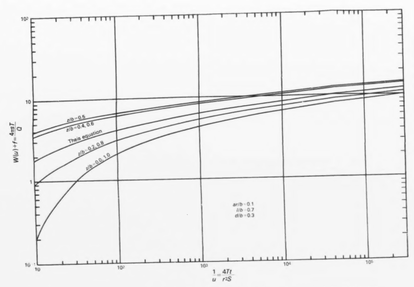

FIGURE 2.2.—Four selected type curves of dimensionless drawdown ($W(u,\beta)$) versus dimensionless time ($1/u$) for constant discharge from a partially penetrating artesian well. The curve for the Theis equation shows the equivalent type curve for a fully penetrating well.

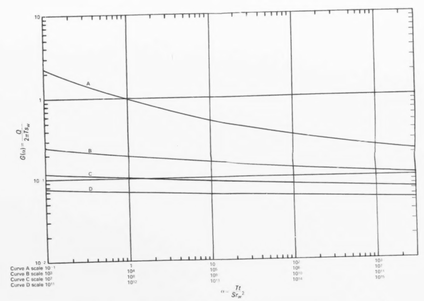

FIGURE 3.2.—Type curve of dimensionless discharge ($G(u)$) versus dimensionless time (u) for constant drawdown in an artesian well. After Lohman (1972).

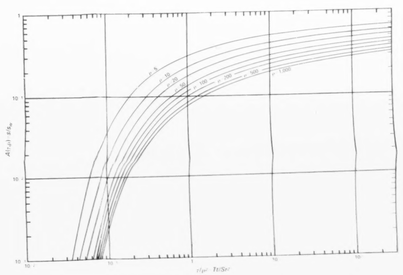

FIGURE 3.3.—Type curves of dimensionless drawdown ($A(u,\rho)$) versus dimensionless time ($1/u$) for constant drawdown in a well in a unconfined aquifer.

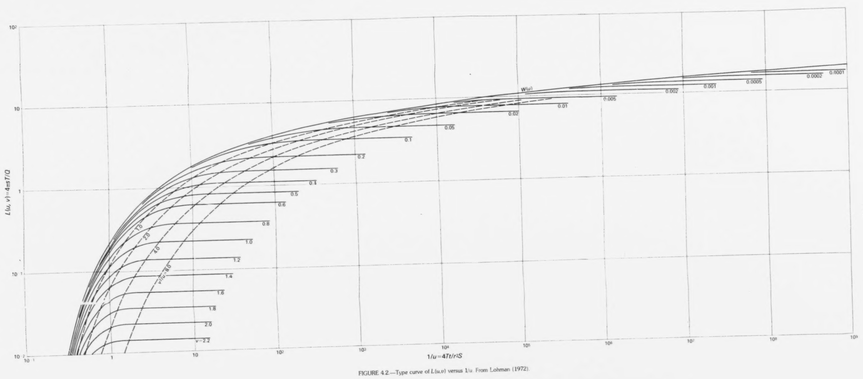

FIGURE 4.2.—Type curve of $L(u,v)$ versus $1/u$. From Lohman (1972).

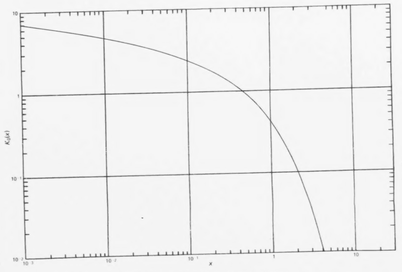

FIGURE 4.3.—Type curve of the Bessel function $K_0(x)$ versus x.

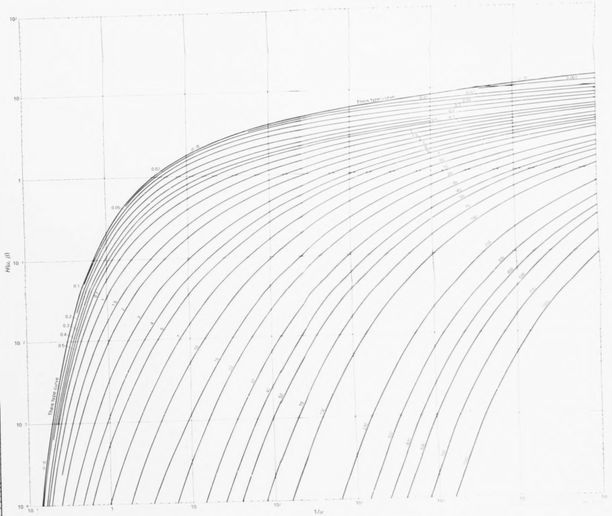

FIGURE 5.2.—Dimensionless drawdown ($Fig.\beta$) versus dimensionless time ($1/u$) for a well fully penetrating a leaky artesian aquifer with storage in leaky confining beds.

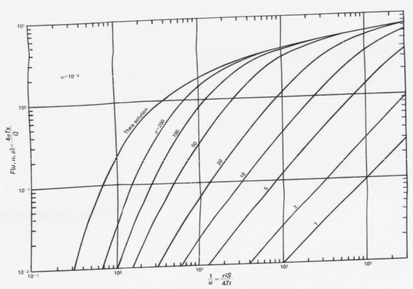

FIGURE 6.2.—Eight selected type curves of dimensionless drawdown ($W(u,\beta)$ or F) versus dimensionless time ($1/u$) for constant discharge from a partially penetrating well in a leaky artesian aquifer. The curve for $r/B=0$ shows the equivalent type curve for a nonleaky aquifer.

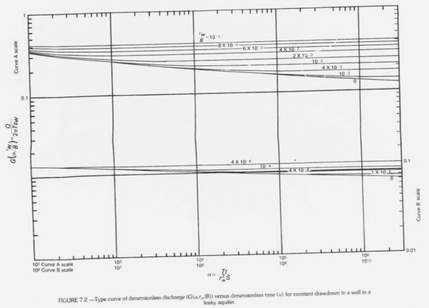

FIGURE 7.2.—Type curve of dimensionless discharge ($G(\alpha,r_e/B)$) versus dimensionless time (α) for constant drawdown in a well in a leaky aquifer.

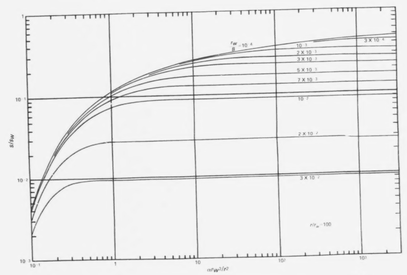

FIGURE 7.3.—Ten selected type curves of dimensionless drawdown versus dimensionless time for constant drawdown in a well in a leaky aquifer.

FIGURE 8.3.—Eight selected type curves of $F(u,\alpha,\beta)$ for $\alpha=10^{-4}$, and the Theis solution, versus $1/u$.

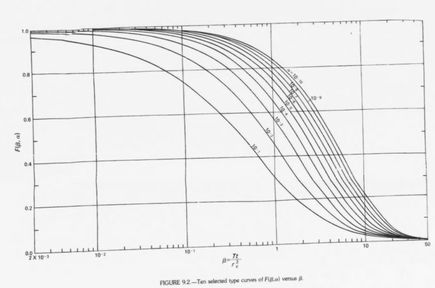

FIGURE 9.2.—Ten selected type curves of $F(\beta,\alpha)$ versus β.

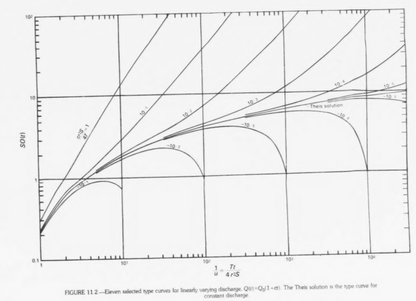

FIGURE 11.2.—Eleven selected type curves for linearly varying discharge. ($Q(t)=Q_0(1-\sigma t)$. The Theis solution is the type curve for constant discharge.

TYPE-CURVE SOLUTIONS FOR CONFINED GROUND-WATER FLOW TOWARD A WELL IN AN INFINITE AQUIFER